AI/ML-Image Recognition, Classification And OCR

Prof. Preet Kanwal
dept. Of Computer Science and Engineering
PES university
Bangaluru, India
preetkanwal@pesu.edu

Ganesh L Kalkote
dept.Of Computer Science and Engineering
PES university
Bangaluru, India
ganikalkote1234@gmail.com

Vivek M
dept.Of Computer Science and Engineering
PES university
Bangaluru, India
viki49081@gmail.com

Inchara C R
dept.Of Computer Science and Engineering
PES university
Bangaluru, India
incharacr.001@gmail.com

Sharath S
dept.Of Computer Science and Engineering
PES university
Bangaluru, India
sharathgowda031@gmail.com

***Abstract*— In recent year, with the speedy development in the digital contents identification, automatic classification of the images became most challenging task in the fields of computer vision. Automatic understanding and analyzing of images by system is difficult as compared to human visions. Several research have been done to overcome problem in existing classification system, but the output was narrowed only to low level image primitives. However, those approach lack with accurate classification of images. In this paper, our system uses deep learning algorithm to achieve the expected results in the area like computer visions. Our system present Convolutional Neural Network (CNN), a machine learning algorithm being used for automatic classification the images. Our system uses the data set as a bench mark for classification of grayscale images. The grayscale images in the data set used for training which require more computational power for classification of images. By training the images using CNN network we obtain the 98% accuracy result in the experimental part it shows that our model achieves the high accuracy in classification of images.**

I. INTRODUCTION

CONVOLUTIONAL NEURAL NETWORK

Image Classification is the process of grouping areas of an image into a number of classes or categories that represent similar features. The process produces "thematic maps" based on the original image or data. Image classification is a major problem in computer vision. An image consists of a set of pixels; pixels from an image are placed with a pooling layer. For image classification, it requires more processing and calculations.

Image classification is where a computer can analyze an image and identify the ‘class’ the image falls under. A class is essentially a label, for instance, ‘car’, ‘animal’, ‘building’ and so on.

CNN uses the varieties of pooling layer for extraction of the feature and applying those features to test images to predict the output. There exists several approaches for image classification, compared with other approaches CNNs provide better accuracy. CNN has 3 different layers: Input layer, Hidden layer, Output layer. Image pixel values (input) are given as input to the input layer along with weights and biases (using relu for non-linearity). The hidden layer may be Convolutional, pooling, the output layer to classify the image to which class it belongs to. The Relu a model that uses it is easier to train and often achieves better performance.

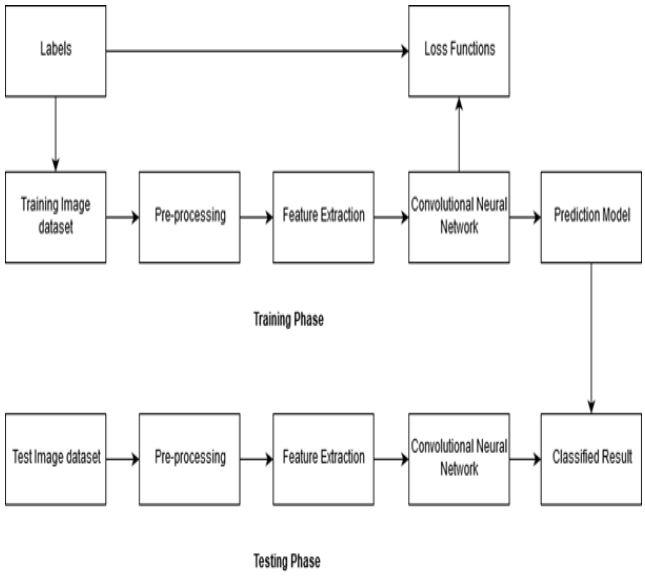


Fig 1: Process of Image Classification.

OCR

Deals with recognizing different kinds of characters in an image. Characters in an image are recognized, transferred into a machine-readable format. Mostly used as a sort of data entry.

Tesseract

Tesseract is an optical character recognition (OCR) system. It is used to convert image documents into editable/searchable PDF or Word documents. Tesseract is considered one of the most accurate open source OCR engines currently available and It will read and recognize the text in images, license plates, etc.

- Tesseract is very powerful OCR-engine and may deliver quite high recognition results.
- High Accuracy.
- Processing of OCR information is fast.
- Easy to install and Implement.

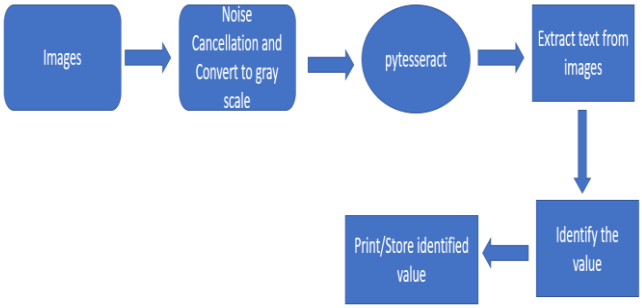


Fig 1.1: process of Pytessract ocr.

II. IMAGE CLASSIFICATION METHODS

Following steps for image classification includes:

a. *Image Acquisition:* Requires the pictures for image processing.
b. *Image Pre-Processing:* At pre-processing transformation of image is done.
c. *Extraction of Feature:* Extracting the required features from the image.
d. *Classification:* Based on extracted features, applying those features to classify the images.

Support Vector Machine (SVM): This method involves a group of hyper planes during a high dimensional space, which is use for classification or regression. The great separation achieved by the hyper plane. SVM uses non-parametric with binary classifier approach and handles more input file efficiently. Accuracy depends on hyper plane selection. This provides the low result transparency.

Artificial Neural Network (ANN): This neural network is a part of AI that works same as human brain. An ANN has varieties of layers. Each layer consists of a group of neurons. By using this method, the performance during the input is extremely fast, but training the images is very slow but prediction will be fast once training is done.

Decision Tree: Decision tree uses a tree-like model of decisions. This method provides information whether to accept or reject the labels at each intermediate (immediate) stage.

Method Name	**Advantages**	**Disadvantages**
SVM	➢ Provides unique solution. ➢ Very efficient than other methods.	➢ Complexity of algorithm is high. ➢ Run slowly.
ANN	➢ Deliver more accuracy. ➢ Very efficient for large dataset.	➢ Cost effective. ➢ Lazy learner. ➢ Slow for training data.
Decision Tree	➢ Easy method as little effort from end users. ➢ Easy to explain.	➢ Splits are very sensitive to training data set. ➢ Error rate is high.

Table 1: *Image classification methods*

III. CNN LAYERS

Conv2d: given that our input would be RGB images which have 3 channels (**RED-GREEN-BLUE**), we specify the number of in_channels as 3. Next we want to apply 12 feature detectors to the images, so we specify the number of out_channels to be 12. Here we use the standard 3 x 3 kernel size.

ReLU: This is the standard Relu activation function, it basically thresholds all incoming features to be 0 or greater. When you apply relu to the incoming features, any number less than 0 is changed to zero, while others are kept the same.

MaxPool2d: This layer reduces the dimension of the image by setting the kernel_size to be 2. What it essentially does is take the maximum of the pixels in a 2 x 2 region of the image and use that to represent the entire region; hence 4 pixels become just one.

Linear: The final layer of our network. It's a standard, fully connected layer that computes the scores for each of our classes — in this case 15 classes

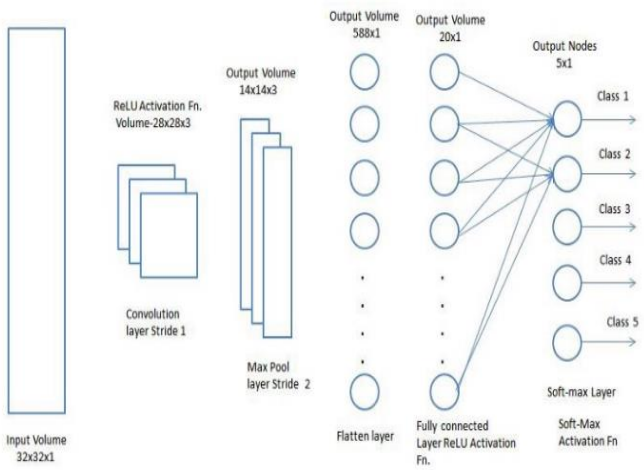


Fig 1.3: Working of CNN.

IV. PERFORMANCE ANALYSIS

Train Accuracy: Training accuracy means that identical images are used both for training and testing. Training accuracy may vary depend upon images and also grouping of images.

Test Accuracy: test accuracy represents that the trained model identifies independent images that were not used in training.

```
Epoch: 0 Train Loss: tensor(31.0109) Train Accuracy: 0.19727891156462585 Test Accuracy: 0.2577082374597331
Epoch: 1 Train Loss: tensor(6.8553) Train Accuracy: 0.2925170068027211 Test Accuracy: 0.3515876668200644
Epoch: 2 Train Loss: tensor(3.5846) Train Accuracy: 0.3881552621048419 Test Accuracy: 0.12517257248044178
Epoch: 3 Train Loss: tensor(2.3190) Train Accuracy: 0.47098839535814324 Test Accuracy: 0.2991256327657616
Epoch: 4 Train Loss: tensor(1.3142) Train Accuracy: 0.6086434573829532 Test Accuracy: 0.12103083294983893
Epoch: 5 Train Loss: tensor(1.5521) Train Accuracy: 0.5938375350140056 Test Accuracy: 0.3506672802577082
Epoch: 6 Train Loss: tensor(0.8831) Train Accuracy: 0.7262905162064826 Test Accuracy: 0.45881270133456054
Epoch: 7 Train Loss: tensor(0.5362) Train Accuracy: 0.8379351740696278 Test Accuracy: 0.3617119190059825
Epoch: 8 Train Loss: tensor(0.3905) Train Accuracy: 0.8731492597038816 Test Accuracy: 0.2485043718361712
Epoch: 9 Train Loss: tensor(0.5632) Train Accuracy: 0.8259303721488596 Test Accuracy: 0.4252185918085596
Epoch: 10 Train Loss: tensor(0.5354) Train Accuracy: 0.8479391756702681 Test Accuracy: 0.43028071790151186
Epoch: 11 Train Loss: tensor(0.3654) Train Accuracy: 0.9071628651460584 Test Accuracy: 0.4132535664979291
Epoch: 12 Train Loss: tensor(0.1815) Train Accuracy: 0.9483793517406963 Test Accuracy: 0.4928670041417395
Epoch: 13 Train Loss: tensor(0.1932) Train Accuracy: 0.9395758303321329 Test Accuracy: 0.38196042337781866
Epoch: 14 Train Loss: tensor(0.0905) Train Accuracy: 0.9795918367346939 Test Accuracy: 0.4413253566497929
```

Fig 1.4: train and test Accuracy using CNN.

V. PY-TESSERACT

Step 1 *Loading and pre-processing the images:* Loading all the images into the memory and converting it to binary grayscale.

Step2 *Using tesseract convert image to text:* Using pre-processed data, convert it into text using image_to_string.

Step 3 *extracting amount value from text using grammar:* Converted text given as input to the grammar to extract the amount value.

Step 4 *storing the predicted values into the excel file.*

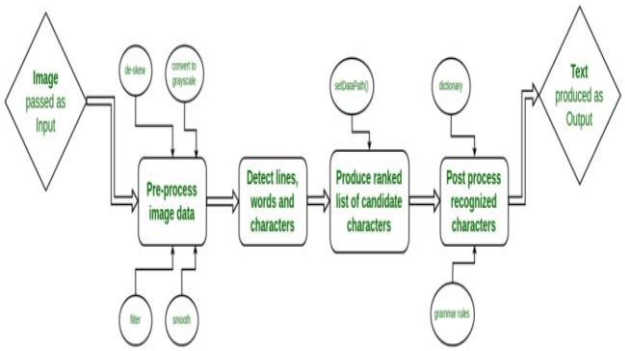


Fig 1.5: Working of Pytesseract.

VI. RESULTS OF THE WORK DONE.

OBJECTIVE 1:

Image classification using CNN

There are 15 classes/labels of an activity type.

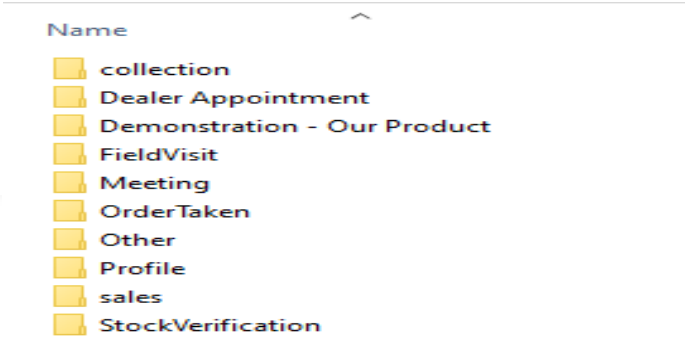


Fig 1.6: Classes/labels of images

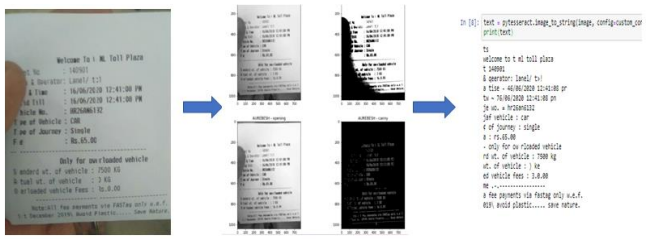
45689	45689 Activity_6c50a58d-76fb-4b46-9e0c-bdac600fa5fb.jpg	93515	106825	1	93515	50320 Dealer	Collection	1 Sales	FALSE
45690	45690 Activity_b5ea55e9-6dc1-433c-9bfd-6995b3f623c1.jpg	93517	106827	1	93517	50369 Retailer	Sales	1 Sales	TRUE
45691	45691 Entity_1262d587-4a8b-4a81-873b-240044c7f940.jpg	93519	106828	1	93519	50269 Dealer	Profile	1 Sales	FALSE
45692	45692 Activity_01507f3e-b150-4c83-b0dc-2838c850b777.jpg	93520	106829	1	93520	50269 Dealer	Sales	1 Sales	TRUE
45693	45693 Activity_1c86e578-ff87-45b6-aa7c-b79b787a51e1.jpg	93521	106830	1	93521	50315 Dealer	Collection	1 Collection	TRUE
45694	45694 Activity_5650f370-1eca-4c8f-82b0-fdcd42fe1d2c.jpg	93522	106831	1	93522	50317 Branch Offi	Other	2 Stock Verification	FALSE
45695	45695 Activity_190c2ec4-2ee2-412d-93d6-4d0184e481b5.jpg	93522	106832	2	93522	50317 Branch Offi	Other	2 Stock Verification	FALSE
45696	45696 Activity_753f7a41-77a9-479a-bd8e-5163ea6009d9.jpg	93523	106833	1	93523	50312 Dealer	Sales	1 Sales	TRUE
45697	45697 Entity_ba68168b-4ecf-40c8-93c5-ee3db6c94ac6.jpg	93524	106836	1	93524	50375 Retailer	Profile	1 Sales	FALSE
45698	45698 Activity_06a19892-4944-444e-b304-ba18ea03ef9e.jpg	93525	106838	1	93525	50300 Dealer	Sales	1 Sales	TRUE
45699	45699 Activity_6eff1432-892b-4438-9822-3db032c285a1.jpg	93526	106839	1	93526	50327 Dealer	Dealer App	1 Sales	FALSE
45700	45700 Entity_d9d0a7f1-f5d1-429a-846c-b23461dbce58.jpg	93527	106842	1	93527	50260 Dealer	Profile	1 Profile	TRUE
45701	45701 Activity_7ea48a34-c22f-4e1b-8dfc-23202c6448ac.jpg	93529	106844	1	93529	50268 Dealer	Sales	1 Sales	TRUE
45702	45702 Activity_8fcdd8a9-9c81-4289-bdae-d2ae64cb8859.jpg	93532	106845	1	93532	50279 Dealer	Sales	1 Profile	FALSE
45703	45703 Entity_62a0a2d2-9fea-4adb-9c89-5325d67a265d.jpg	93533	106846	1	93533	50343 Retailer	Profile	1 Sales	FALSE
45704	45704 Activity_64da257a-4577-4209-9647-371d8f49070f.jpg	93534	106848	1	93534	50263 Dealer	Field Visit	1 Sales	FALSE
45705	45705 Activity_9fc8c885-832a-4b20-b4bc-984395f9811f.jpg	93536	106849	1	93536	50317 Branch Offi	Other	1 Stock Verification	FALSE
45706	45706 Entity_c95dec92-e6ff-49b4-ae5f-3492db700cc6.jpg	93537	106850	1	93537	50262 Retailer	Profile	1 Sales	FALSE
45707	45707 Activity_14b31b57-34ac-44b8-8c11-dfd2402661fc.jpg	93538	106851	1	93538	50262 Retailer	Collection	1 Sales	FALSE
45708	45708 Activity_3aee43dd-329e-4624-aeb8-b8bd4b3cab07.jpg	93539	106852	1	93539	50312 Dealer	Sales	1 Collection	FALSE
45709	45709 Activity_a37de2ea-e887-4c2d-85fa-8e60519177f.jpg	93540	106853	1	93540	50379 Dealer	Sales	1 Collection	FALSE
45710	45710 Activity_cd14d44e-61ed-4a14-93c6-48c16742d366.jpg	93542	106854	1	93542	50325 Dealer	Sales	1 Sales	TRUE
45711	45711 Entity_99bd1fc4-da1e-462e-af48-22d660c6ff2a.jpg	93544	106855	1	93544	50353 Retailer	Profile	1 Sales	FALSE
45712	45712 Activity_aabebf12-2f10-4d0e-ad7e-3ae8e49576c1.jpg	93545	106856	1	93545	50353 Retailer	Sales	1 Sales	TRUE
45713	45713 Activity_1a0e3f64-282b-4847-9bed-377566028878.jpg	93547	106858	1	93547	50353 Dealer	Sales	1 Sales	TRUE
45714	45714 Entity_2d891724-4663-42d9-8b6c-5071c70d9ac5.jpg	93549	106859	1	93549	50355 Dealer	Profile	1 Sales	FALSE
45715	45715 Entity_51290816-165c-4150-9cee-e1c2026b4608.jpg	93550	106860	1	93550	50274 Retailer	Profile	1 Sales	FALSE
45716	45716 Activity_91f8aec1-87a0-4ec3-b658-4a7e935b7226.jpg	93553	106867	1	93553	50350 Dealer	Collection	1 Sales	FALSE
45717	45717 Activity_ef7d950f-2045-417d-aaa7-30ba51f213c7.jpg	93555	106868	1	93555	50261 Dealer	Collection	1 Sales	FALSE
45718	45718 Activity_ceaf17d6-ae76-4940-ae6a-810011c21e81.jpg	93556	106869	1	93556	50264 Dealer	Sales	1 Collection	FALSE

Fig 1.7: Activity of an employee prediction and comparing it with existing Activity type.

OBJECTIVE 2:

Extract values from images using ocr (Pytesseract)

Preprocessing of images



Original image after noise cancellation, threshold and gray scale images converting to text

Fig 1.8: *Noise removal from images*

ImageFileName	Amount
Activity_cf861910-8ff8-44ec-a7a2-1996ecd50748.jpg	65
Expense_037a4d8c-c7d7-42f3-9bb4-7f888dbd4857.jpg	0
Expense_0b668d71-52b8-4907-9e29-2bdb10e4e183.jpg	0
Expense__00b80a48-4dd8-4629-8ee6-a095dfa5bb90.jpg	90
Expense__0187d55a-1bf6-4086-ba1a-d38e941598ea.jpg	0
Expense__01a40037-9b13-4007-984b-06d367d897ec.jpg	0
Expense__01ab464e-0782-41ce-890e-7b1da0b90594.jpg	0
Expense__02d8e71d-7cf3-4364-a8a9-8f96f204f25d.jpg	0
Expense__04653b06-9b61-4f95-9534-63f1e240a4b8.jpg	0
Expense__05090e0e-e240-44db-8616-c1ff6fe98e1d.jpg	0
Expense__0a5eaa07-c9dd-40fc-b530-00ee3bf51510.jpg	0
Expense__0b5617cb-0028-4073-a303-05bd7c9182e3.jpg	0
Expense__0b5f99f6-6e02-4122-b7d6-68c1d6f6ab25.jpg	0
Expense__0ba93fa4-ccfa-4fe5-8a4b-993d23231ba9.jpg	77,55
Expense__0bb9e874-cdae-41e0-988b-f628513e5ad3.jpg	0
Expense__0bd322b7-b460-4140-b912-eaffe21e0df9.jpg	65
Expense__0bdb6edc-b3e8-47d3-ab7c-0053b37f85b1.jpg	0
Expense__0bdf99b3-c795-4e8e-95ab-9b8ebe6e4ce1.jpg	0
Expense__0bf52a75-f767-47de-b5d4-5679f9a7b71c.jpg	0
Expense__0c7ab50b-a8a3-4c51-9494-7aef5e829c84.jpg	40
Expense__0ca92a25-98da-42b0-aae7-8802dc53cf1e.jpg	70
Expense__0ce8eede-c0a2-4275-a2c9-c03aaf0c3743.jpg	0
Expense__0cf77879-6272-48ba-850f-55355e6a1b32.jpg	0
Expense__0d14bc61-8841-4a7e-b7ce-fe07cb008cf0.jpg	0
Expense__0d43e776-0901-43d0-bd07-c79256f24c90.jpg	0
Expense__0d6916c3-cf9d-411e-927d-87f0ab1a247e.jpg	0
Expense__0d774831-8967-41fd-ac64-358cbd9aa623.jpg	25,00
Expense__0db544cb-ed65-4985-b44e-6eb0feed3b98.jpg	0
Expense__0dd2788c-afb9-4d80-a137-1c6a6327db2c.jpg	0

Fig 1.7: *Extracted values from images are stored in excel file using pytesseract.*

VII. CONCLUSION

In this project the main focus is on image classification and recognition. The model used for image classification is CNN (convolution neural network) and the tool used for recognition of text in image is Pytesseract. There are two objectives: one is about image classification and the second is image recognition. In the first objective i.e. image classification, there are about 70 to 80 images as dataset, these images have been trained and tested before the whole actual images were classified using CNN model to extract the features of an images, in the later stage after obtained the best accuracy model result from test and trained images, the whole images are given as input into that model for classification of images. Next in the second objective i.e. image recognition, using Pytesseract as a tool to identify the characters and symbols present in the given set of images and saves the result in excel file.

REFERENCES

[1] https://www.geeksforgeeks.org/difference-between-ann-cnn andrnn/#:~:text=ANN%20is%20considered%20to%20be,compatibility%20when%20compared%20to%20CNN.&text=Facial%20recognition%20and%20Computer%20visio n.

[2] https://towardsdatascience.com/introduction-to-hidden-markov-modelscd2c93e6b781

[3] https://perso.liris.cnrs.fr/cwolf/papers/cbmi2009-marc.pdf

[4] https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neuralnetworks-the-eli5-way-3bd2b1164a53

[5] https://towardsdatascience.com/https-medium-com-pupalerushikesh-svmf4b42800e989#:~:text=SVM%20or%20Support%20Vector%20Machine,separates%20the%20data%20into%20classes

[6] https://medium.com/analytics-vidhya/fauna-image-classification-using-convolutionalneural-network-30df9e25a010

[7] https://ijcrt.org/papers/IJCRT2109326.pdf