Pakistan Journal of Engineering and Technology, PakJET
Multidisciplinary & Peer Reviewed
Volume: 4, Number: 2, Pages: 144- 148, Year: 2021

Tourist Guide via Image Processing Techniques

Rabia Noor Enam[1], Muhammad Tahir[2], Syed Muhamad Nabeel Mustafa[3], Rehan Quresh[4], Hasan Shahid[5]

[1]Computer Engineering Department, Sir Syed University of Engineering and Technology, Karachi, Pakistan
[2,4,5]Software Engineering Department, Sir Syed University of Engineering and Technology, Karachi, Pakistan
[3]Computer Science and Information Technology Department, NED University of Engineering and Technology, Karachi, Pakistan
Corresponding author: Rabia Noor Enam (e-mail: rnenam@ssuet.edu.pk).

***Abstract*-To achieve the goal of identification, image processing and recognition is performed on the actual picture transformation. The amount of information included in an image is enormous because it is a two-dimensional space. Neural network image recognition is a new type of picture recognition technology developed by modern computer innovation. In this paper we have used neural network to implement image-based location identification. Our target dataset was Pakistani tourist places images. The proposed methods serve as a guide for travelers or tourists who are unfamiliar with a country. They simply need to point the camera at any historical or well-known location to learn more about it. Using the image data, we have evaluated our proposed model's predictive performance. By including more hidden layers in the convolution neural network, it is feasible to improve the network's training speed and reliability by having a large amount of data. However, 80% of the dataset is used for training and the remaining 20% is used for testing in our studies. The preliminary findings have shown that using a neural network to detect photos is both successful and practical We have compared our results with another comparable work done previously, our results show a similar result but with only 1/3rd amount of data set used to train the model.**

***Index Terms*-- Convolutional Neural Network (CNN), Image Processing, Image Recognition**

I. INTRODUCTION

With photos becoming the most popular form of material, image classification has been a major focus for businesses looking to speed up their processes. Image recognition is a branch of Computer Vision and Artificial Intelligence that includes a variety of technologies for recognizing and analyzing images in order to automate a task [1]. It's a piece of software that can identify places, people, objects, and a number of other items in a photograph and draw conclusions from them after analyzing them. The development of a reliable model to perform accurate place recognition is still a challenge today [2]. Many have underestimated the complexity of this problem as place recognition is a process that can be easily done by the human eyes. However, for a computer model to predict the place correctly is a big challenge. One of the problems faced in place recognition of similar looking places. These places have similar features due to which it gets difficult to predict the place correctly. Another challenge faced is selecting the unique features of the place to distinguish between places.

Machine learning algorithms (MLAs) are strong tools for solving problems in a wide range of fields, including natural language processing, data mining, speech recognition, and image recognition. Deep learning [3] is an inexorably well-known subset of machine learning. Deep learning is defined as deep neural networks with several nonlinear layers that learn features from input using a general-purpose learning technique rather than being built by humans [4]. Deep learning models are created using neural networks, a machine learning technique based on the principles of biological brain networks' design and operation. Neural networks for image recognition have progressed from simple chain-like models to architectures with numerous wire channels thanks to a lot of manual work [5]. This idea was conceived when Warren McCulloch and Walter Pitts [6] tried to imitate the actions occurring in the human brain. A neural network receives data and connects neurons in each layer to neurons in the next layer. Along these molecules, data travels from the input layer to the output layer. Every node in the network performs a simple mathematical calculation. After that, it sends its data to all of the connected nodes. The model then makes a prediction. The weights are changed to find patterns so that better forecasts can be made. The neural network learns the patterns on its own, sparing the user from specifying the patterns [7]. The construction of hierarchical representations in a convolutional neural network, where different layers represent different levels of visual information, can simulate this compositional behavior in computer vision [8]. Multiple cascaded convolution kernels with deep learning applications are embedded by Convolutional neural networks, in artificial intelligence.

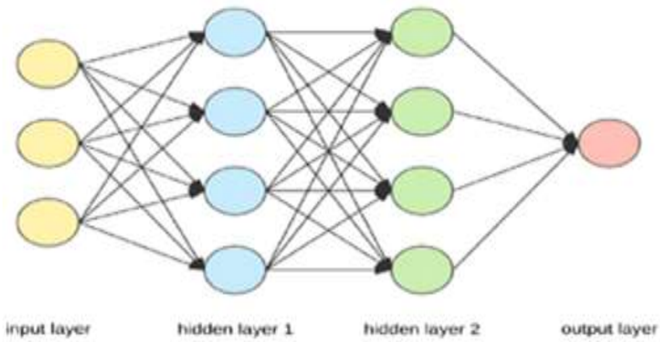

FIGURE 1: Convolutional Layers of Neural Networks

Convolutional Layer computer builds more unique concepts [9]. With architectures that include numerous layers that are taught, Convolutional Neural Networks (CNN) techniques are often used. Visual cortex, a visual process in the brain, was inspiration behind the design of a CNN. In receptive fields, which are small, overlapping sub-regions of the visual field, the visual cortex has a large number of cells that sense light. The receptive fields of more advanced cells are larger, and they act as local filters across the input space. The convolution layer in a CNN performs the same function as the visual brain's cells. As shown in Fig. 1, the network is trained in two stages: forward and reverse. The major goal of the forward stage is to represent the input image using the current parameters in each layer (weights and bias). Image and pattern identification, speech recognition, natural language processing, and video analysis are all disciplines where CNNs are applied. Convolutional neural networks are gaining popularity for a variety of reasons. Feature extractors are built in standard pattern recognition models. During the training phase, the weights of the convolutional layer used for feature extraction and the fully connected layer used for classification are determined in CNNs [10].

For this project, we used Python. We utilized Keras [11] as a framework, which is a high-level neural network package written in Python that is designed to make developing deep learning networks easier. Keras, on the other hand, cannot function without a backend for low-level operations. As a result, we installed TensorFlow [12], a Google-developed software package. Pixels are used by computers to view images. Pixels in images are usually connected; a group of pixels can represent an image's edge or any other pattern [13]. This is how Convolution identifies photos. The computer, for example, looks for characteristics of the basic level. The trunk or large ears are examples of such traits in human thinking. These features, on the other hand, are borders or curvatures for the computer. The computer then creates additional unique notions by grouping convolutional layers together.

II. LITERATURE REVIEW

In [14], authors have demonstrated with the help of CNN with kaggle, Flavia and MEW2012 datasets to recognize the specie of leaf. They had seven codes with fully-connected layers of CNN, that only works with numerical shapes of leaves, to compare it with using images. Authors in [15], with the help of CNN, determines the face recognition technology. LeNet5 architecture was used for analysis. LeNet5 is a CNN classic structure, which is used in the recognition of handwritten fonts. Author in [16] proposes an optimized CNN model for mage recognition. The work is done in two stages. First by selection of target region and then by optimization by enhancement weight-based model. Author in [17] trains a designed CNN model on a dataset of 48×16-pixel resolution taken from coarse meshes. The trained model then predicts the image-based topologies based of the fine mashes. CNN models are created in [18] to assess their performance on picture recognition and detection datasets. On the MNIST and CIFAR-10 datasets, the technique is implemented and its performance is evaluated. Authors in [19] implemented deep learning on hand written character recognition. The performance evaluation for CNN wad conducted on MNIST database and the real-world handwritten character database. In [20], authors improved the microscope by powering it by CNN image recognition algorithm. The architecture classifies epidemic pathogen with five deep learning phases. Authors in [21] have autopsied the CNN and algorithm and have presented their results on MNIST database. This research [22], offers a transfer learning-based Deep Convolutional Neural Networks (CNN) model for image recognition. To learn feature selection, a Deep CNN system was pre-trained on the enormous ImageNet dataset of 14 million images and 1000 classes. By adding two normalization procedures to two of the layers, [23], offers a modified Convolutional Neural Network (CNN) architecture. The network was accelerated via batch normalization, which is a normalizing procedure. In the fully connected layer of CNN, the CNN architecture was utilized to extract distinguishing face features, and the Softmax classifier was used to classify faces. In [24] , using the image mining techniques, the authors have done the study work on representative images or models of sightseeing places by analyzing the photos uploaded on Flickr. In their work the authors have used a pre-trained model, Inception v3 model, which is created by training the ImageNet data. Our work in this paper is quite similar to this one therefore we have compared our work with this work in our Results section.

III. METHODOLOGY

The picture recognition model is created with Keras and the Convolution Neural Network (CNN), which has three layers: convolution, pooling, and flattening, as depicted in Fig 2. Convolution's main goal is to extract features from the input image. By learning image attributes using small squares of input data, convolution preserves the spatial link between pixels. Convolution transfers to another pixel and this process is repeated until all of the pixels in the image have been retrieved. This procedure is depicted in Fig 3. The feature map is another name for the obtained matrix.

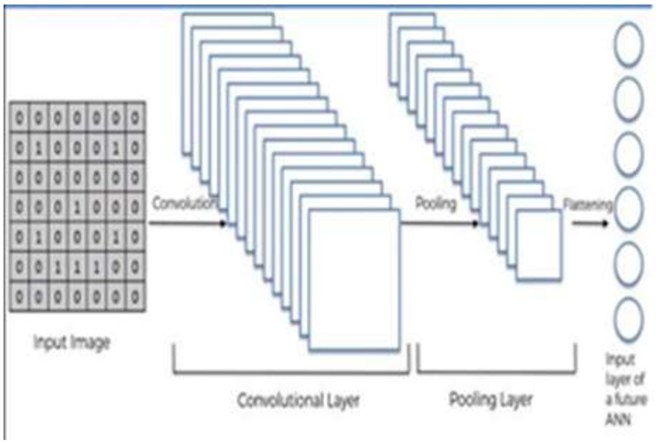

FIGURE 2: Structure of Convolutional Networks

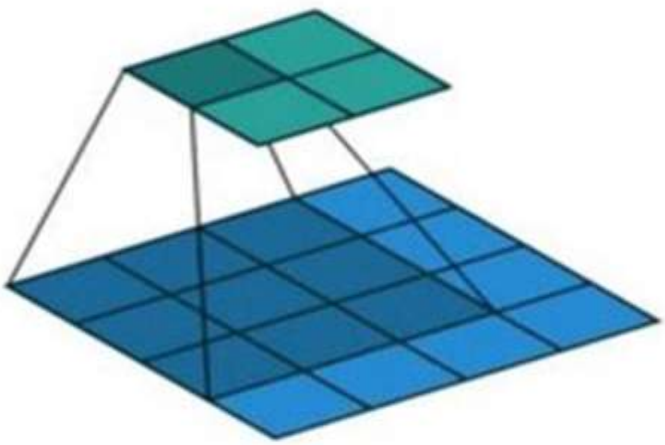

FIGURE 3: Visualization of Image recognition

Models using an activation function can account for nonlinear relationships. After each Convolution process, we applied the ReLU- Rectified Linear Unit activation function. In the last few years, ReLU has gained a lot of traction. Because, the function y = max(x,0), is implemented by ReLU, this layer's input and output sizes are the same. It improves the decision function's and overall network's nonlinear qualities without impacting the convolution layer's receptive fields [10].

if $x < 0$, $R(x) = 0$ and if $x >= 0$, $R(x) = x$.

The graphical representation is shown in Fig 4. It avoids and corrects the vanishing gradient problem since the gradient is always high (equal to 1) and does not saturate when the neuron activates. However, one of its most significant drawbacks is that it can only be used within the hidden layers of a neural network model. As we are dealing with the Classification problem, we have utilized a Softmax function to compute the probabilities for the output layers.

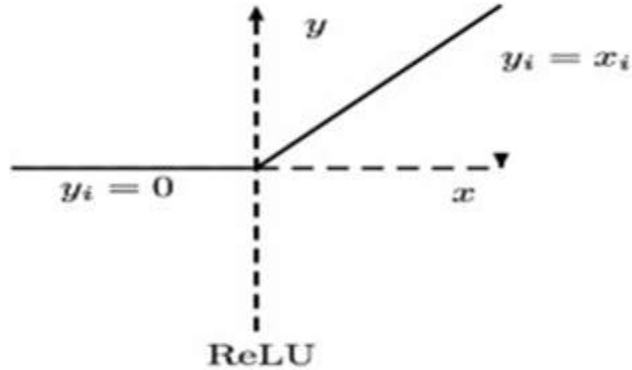

FIGURE 4 Graphical Representation of ReLU

With a 2×2 window size, Fig 5, we utilized the maxPooling function, which takes the largest element from the corrected feature map within that window [25].

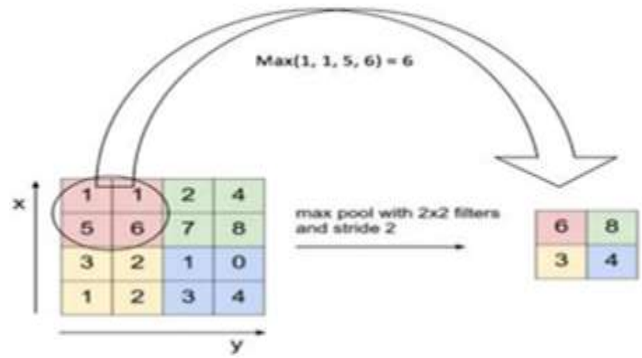

FIGURE 5: Feature Map of Pooling Layer

Following Pooling, follows Flattening, which, as the name implies, transforms the matrix into a linear array that can be fed into the nodes of our neural network. All nodes from previous layers are connected to nodes in the current layer in a dense laye. We then assembled our model by connecting all of the convolution network layers to a neural network. The working of Flattening layer is shown in Fig 6.

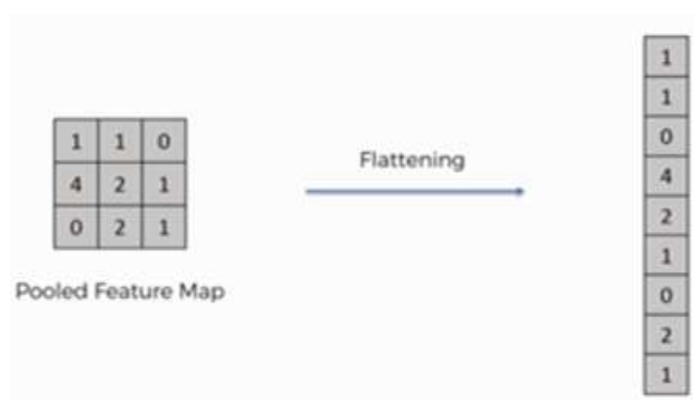

FIGURE 6: Feature Map of Flattening Layer

The model is built using two parameters: optimizer and loss. The learning rate is controlled by the optimizer. As our optimizer, we'll be utilizing RMSprop. For the most part, it's an excellent optimizer to use. The The RMSprop optimizer limits vertical oscillations [26]. As a result, we may be able to increase our learning rate, which will allow our algorithm to take more horizontal steps and converge faster. The learning rate determines how soon the ideal weights for the model are computed. A slower learning rate produces more accurate weights (to a point), but it also takes longer to compute them. We selected 'categorical crossentropy' [27] as our loss function, which is a popular loss function for multiclass classification issues. Mathematically it can be defined as

$$L(y,\hat{y}) = -\sum_{j=0}^{M}\sum_{i=0}^{N}(y_{ij} * log(\hat{y}_{ij}))$$

Where,

- $\hat{y}$ is the predicted value
- M is the number of classes
- N is the number of observations
- y is the binary indicator.

The distribution of the predictions will be compared to the true distribution using categorical cross entropy. The correct class's probability is set to 1 and the remaining classes' probabilities are set to 0. To put it another way, true class is encoded as a one-hot vector, and the less the loss, the closer the model's outputs are to that vector.

If there are a large number of photos for training, an image classification model has a good chance of performing well. While training, a big amount of data is required for training photos pertaining to the classes you are working on, so we used a JavaScript code and a little Python to get all of the image dataset from Google. Using Google Chrome, a simple JavaScript code is built on the JavaScript console, which captures all the URLs of photos that we searched for on Google. The URLs of the photos are then supplied into the python code, which uses the requests module to download all of the images' content. The dataset is divided into 6000 (750 photos for each location) images for training, 1600 (200 images for each location) images for testing the model's performance during training, and 400 (50 images for each location) images for validation.

IV. RESULTS

We ran a series of tests with our collected dataset of 8000 photos of Pakistani tourist places, to see how well the proposed methodology performed. Our goal is to evaluate our model's predictive performance using image data. As a result, 80% of the dataset is used for training and the remaining 20% is used for testing in our studies. The best classification results came from the complete dataset, as appeared in Fig 7. We have shown our result in terms of accuracy/loss for training and validation while training. In Tab. I, the results of [24] can be seen, where, similar to our work, different places of interests were taken and for each place a different result of accuracy has occurred which in total gives 27.93% of accuracy overall.

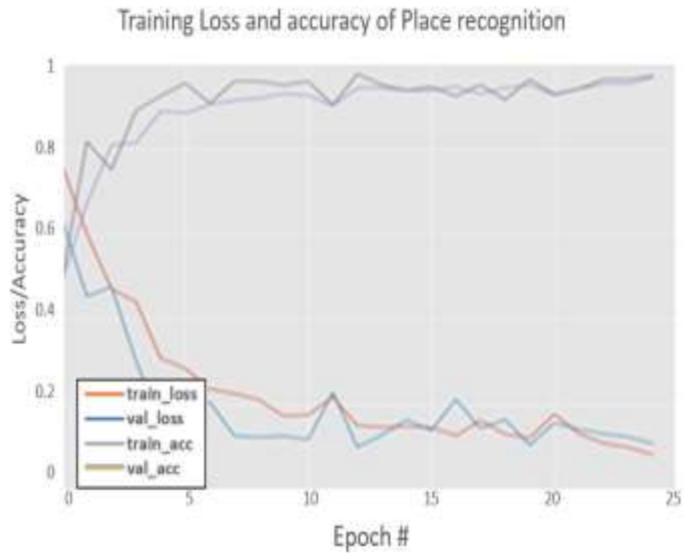

FIGURE 7: Set of Experiments using Collected data set

TABLE I. Accuracy ratio result from Collected data set [24]

Predicted categories	Total	TRUE	FALSE	Accuracy ratio (%)
Palace	1357	1070	287	78.85
Bell cote, bell cot	791	25	766	3.16
Plate	752	750	2	99.73
Restaurants	672	450	222	66.96
Toyshop	636	201	435	31.6
Grocery store, grocery,	628	317	311	50.48
Cinema, movie theater,	605	37	568	6.12
Lakeside, lakeshore	598	168	430	28.09
Patio, terrace	516	140	376	27.13
Cab, hack, taxi, taxicab	501	84	417	16.77
Tile roof	496	415	81	83.67
Hot pot, hotpot	434	290	144	66.82
Pier	432	216	216	50
Menu	431	134	297	31.09
Stage	428	185	243	43.22
Prison, prison house	427	16	411	3.75
Traffic light, traffic signal,	406	139	267	34.24
Monastery	394	2	392	0.51
Total	38,691	10,807	27,884	27.93

V. CONCLUSION

In this work, a neural network is used to implement image-based location identification. By including more hidden layers in the convolution neural network, it was feasible to improve the network's training speed and reliability by having a large amount of data. Training time is decreased in half thanks to adaptive learning rate modification. The preliminary findings indicate that employing a neural network to detect photos is both successful and practical while using a reduced amount of data set. As 80% of the dataset is used for training and the remaining 20% is used for testing. Our results have shown the training and validation accuracy reaches to almost 80% after 25 Epochs.

REFERENCES

[1] Fang, Wei, et al. "A method for improving CNN-based image recognition using DCGAN." *Computers, Materials and Continua* vol.57, no.1, pp. 167-178, 2018.

[2] Zerdoumi, Saber, et al. "Image pattern recognition in big data: taxonomy and open challenges: survey." *Multimedia Tools and Applications* vol.77, no.8, pp. 10091-10121, 2018.

[3] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[4] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature* vol.521, no.7553, pp. 436-444, 2015.

[5] Xie, Saining, et al. "Exploring randomly wired neural networks for image recognition." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019.

[6] Ksenia Sorokina "Image Classification with Convolutional Neural Networks" Nov 20, 2017

[7] LI Huanliang "The Research of Intelligent Image Recognition Technology Based on Neural Network"

[8] Hu, Han, et al. "Local relation networks for image recognition." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019

[9] Giuseppe Amato, Fabio Carrara, Fabrizio Falchi, Claudio Gennaro, Gabriele Lagani "Hebbian Learning Meets Deep Convolutional Neural Networks" International Conference on Image Analysis and Processing ICIAP 2019

[10] Hijazi, Samer, Rishi Kumar, and Chris Rowen. "Using convolutional neural networks for image recognition." *Cadence Design Systems Inc.: San Jose, CA, USA* 2015, pp. 1-12.

[11] Ketkar, Nikhil. "Introduction to keras." *Deep learning with Python*. Apress, Berkeley, CA, 2017, pp.97-111.

[12] Smilkov, Daniel, et al. "Tensorflow. js: Machine learning for the web and beyond." *arXiv preprint arXiv:1901.05350* 2019.

[13] Lee, Sang-hoon, Jae Sung Park, and Nam Ik Cho. "A multi-exposure image fusion based on the adaptive weights reflecting the relative pixel intensity and global gradient." *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018.

[14] Jesus L ´ opez Barrientos ´ "Leaf Recognition with Deep Learning and Keras using GPU computing" , TFG IN COMPUTER ENGINEERING, ENGINEERING SCHOOL (ES), UAB UNIVERSITY (UAB) 2018

[15] Jie Wang, Zihao Li "Research on Face Recognition Based on CNN" IOP Conference Series Earth and Environmental Science, vol.170, no.3, pp.032110, July 2018

[16] Hao, Wu, et al. "Optimized CNN based image recognition through target region selection." *Optik*, vol.156, pp. 772-777, 2018.

[17] Lee, Seunghye, et al. "CNN-based image recognition for topology optimization." *Knowledge-Based Systems, vol.* 198, pp. 105887, 2020.

[18] Chauhan, Rahul, Kamal Kumar Ghanshala, and R. C. Joshi. "Convolutional neural network (CNN) for image detection and recognition." *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*. IEEE, 2018.

[19] Meiyin Wu and Li Chen, "Image recognition based on deep learning," 2015 Chinese Automation Congress (CAC), 2015, pp. 542-546.

[20] Traore, Boukaye Boubacar, Bernard Kamsu-Foguem, and Fana Tangara. "Deep convolution neural network for image recognition." *Ecological Informatics*, *vol.* 48, pp. 257-268, 2018.

[21] Liu, Yu Han. "Feature extraction and image recognition with convolutional neural networks." *Journal of Physics: Conference Series*. Vol. 1087. No. 6. IOP Publishing, 2018.

[22] C. Iorga and V. Neagoe, "A Deep CNN Approach with Transfer Learning for Image Recognition," *2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 2019, pp. 1-6

[23] Coşkun, Musab, et al. "Face recognition based on convolutional neural network." *2017 International Conference on Modern Electrical and Energy Systems (MEES)*. IEEE, 2017.

[24] Kim, D., Kang, Y., Park, Y. et al. Understanding tourists' urban images with geotagged photos using convolutional neural networks. Spat. Inf. Res. 2020. https://doi.org/10.1007/s41324-019-00285-x

[25] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." 2020, *arXiv preprint arXiv:2010.11929*.

[26] Wichrowska, Olga, et al. "Learned optimizers that scale and generalize." *International Conference on Machine Learning*. PMLR, 2017.

[27] Das, Abhishek, Gyana Ranjan Patra, and Mihir Narayan Mohanty. "LSTM based Odia handwritten numeral recognition." *2020 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2020.