Journal of the Nigerian Association of Mathematical Physics
***Volume 54* (January *2020 Issue*), pp119 – 128**
© J. of NAMP

CONVOLUTIONAL NEURAL NETWORKS FOR IDENTIFYINGCONTEMPORARY NIGERIAN WOMEN HAIRSTYLES

Chika O Yinka-Banjo[1,] and Oluwafisayomi Sodeinde[2]

Department of Computer Sciences, University of Lagos, Akoka, Nigeria

Abstract

Convolutional Neural Networks (CNNs) are the state of the art in image classification and recognition tasks. This project explores a relatively novel image classification task involving a dataset of Nigerian Contemporary hairstyles distributed across seven classes. For optimal performance, the transfer learning technique is used with the VGG19 network providing the pretrained convolutional layers. The fully connected layers are optimized for classification of the hairstyles. Results show that the CNN achieves test accuracy of 86% in the classification task. Finally, the CNN is deployed as a web application, design specifications of the web application are presented in the project.

***Keywords*:** Convolutional Neural Networks, Deep Learning, Artificial Intelligence, Neural Networks, Image Preprocessing, Image Classification.

1. Introduction

The appearance of one's hair influences one's general appearance thus, making people go to a great length to ensure that their hair is properly groomed. The hairstyle of a Nigerian woman has evolved overtime and has become a mix of the Nigerian traditional hairstyles and hairstyles from all around the world.

Hairstyling is a form of art and hairstyles can tell a story about its wearer and although they are primarily for beautification purposes, hairstyles can have a lot of cultural significance. Some contemporary Nigerian hairstyles are deeply rooted in the Nigerian culture and can be used to indicate the religious beliefs, the marital status, the social status and grieving [1]. These days a lot of women focus on fixing and braiding their hair but there are styles such as threading (kiko), didi (inverted cornrows) etc. Fixing involves sewing or gluing hair extensions one's hair while braiding involves plaiting or twisting of the hair, sometimes with the use of extensions. Traditionally we have three main methods of hairstyling in Nigeria; cornrow, inverted cornrows and threading, the same hairstyle can be portrayed in the different methods. Different ethnic groups have diverse hairstyles as hairstyles were used then to express where a person is from.

In the African society, hairstyles aren't only meant for the primary purpose of beautification and adornment of the head, they depict a lot of cultural significance and the Nigerian society isn't left out, not only are they used to illustrate where a person is from, they showcase the rich culture of the nation.

Certain hairstyles were used to portray royalty as they were only reserved for and only worn by the royals. Some hairstyles were created to portray the political events that occurred in the country, one of such hairstyles is from the Yoruba people of South-Western Nigeria known as "ojukwu dobale fun gowon"; this hairstyle is about the end of the Nigerian civil war and the surrender of the Biafran army, Hairstyles portray one's religious beliefs as different religious had unique hairstyles ascribed to their practices, marital status showcased by the hairstyles worn as there are various hairstyles for brides and married women. Occupation also played a huge role in the types of hairstyles worn as they were used to represent their occupation [1].

The hair and fashion industry at large are a very big industry which should benefit from digital technologies such as web and mobile apps. More advanced technologies such as artificial intelligence and machine learning can add a great value to the industry in term of consumer products and services. Artificial Intelligence (AI) can be used to optimize the hair and fashion services, technologies like Convolutional Neural Network (CNN) can be embedded in apps. AI and CNNs have led to ground breaking technologies in various fields and have become created a more efficient way of computing hence CNNs can be explored in the hair industry. The goal of this paper is to explore the use of state-of-the-art computer vision technology such as the CNN to build a utility web application that provides an efficient means of classifying Nigerian contemporary hairstyles as well as provide useful information about the hairstyle.

Corresponding Author**:** Chika O.Y., Email: cyinkabanjo@unilag.edu.ng, Tel: +2348033434289, +2348069508415 (OS)

In section 2 we present a brief overview of key related concepts regarding the CNN and the transfer learning technique, as well as motivations for building an image classifier for Nigerian hairstyles. Section 3 presents the design methodology and experimental setup while section 4 presents the results.

2.1 Convolutional Neural Networks (CNN)

Convolutional neural networks are supervised deep learning model which typically consists of three layers; the convolution layer, the pooling layer and the classification layer. There is usually an optional image pre-processing layer before the convolution layer which helps to improve the recognition rate. At the convolution layer which is also known as the detector stage, several convolutions are performed in parallel by running a kernel in strides over the pre-processed image to produce a set of linear activations known as the feature map. Each of the linear activation is then run through a nonlinear activation function, such as the rectified linear activation function. At the pooling layer which is sometimes referred to as subsampling, a pooling function such as MIN pooling or MAX pooling is used to modify the output of the layer further. At the classification layer, the output of the last pooling layer is passed to a fully connected neural network [2].

2.1.1 CNN Architecture

Typical, convolutional neural network architecture consists of an input layer, one or more convolution and pooling layer, a fully connected layer and an output layer.

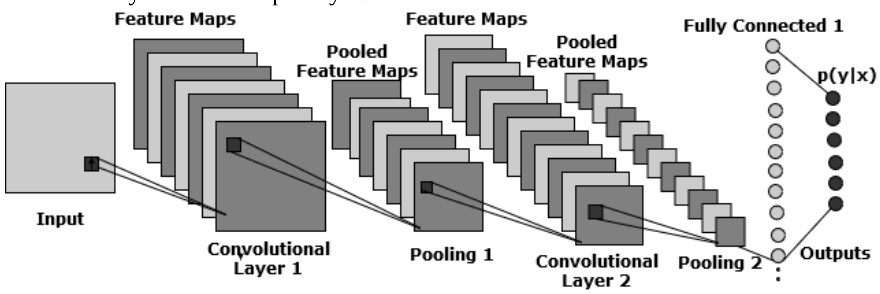

Fig. 1: The convolutional neural network architecture showing the convolution, pooling and fully connected stages [3].

2.1.2 Convolution Operation

The convolution operation is a mathematical operator used to estimate a weighted average over a set of measurements provided by two real valued functions [4]. The convolution operation is usually denoted with an asterisk as shown in Eq. 1 below:

$$s(t) = (x * w)(t) \tag{1}$$

In convolutional network terminology the function x in Eq. 1 is referred to as the input and w is the kernel and the output $s(t)$ is feature map[5], the input x is usually a multidimensional array of data, the kernel w is a multidimensional array of parameters that are adapted by the learning algorithm. These arrays are usually called Tensors [2]. Variants to the convolution operation include unshared convolution and tiled convolution [6].

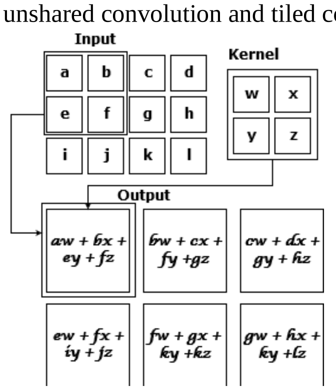

Fig. 2: An example of 2d convolution without kernel flipping [2].

2.1.3 Pooling Operation

A pooling operator (e.g. Min pooling, Max pooling and Average pooling) is used to further modify the output of the convolution layer after several convolutions have been performed in parallel to produce a set of linear activation and each of the linear activation have been run through a nonlinear activation function. It maps a high dimensional image to a low dimensional image; this process is known as down-sampling. The pooling function reduces the representation size thereby reducing the computational and statistical burden on the next layer. Pooling aids making the representation become approximately invariant to small translations of the input. Invariance to translation means that if we translate the input by a small amount, the values of most of the pooled outputs do not change [2].

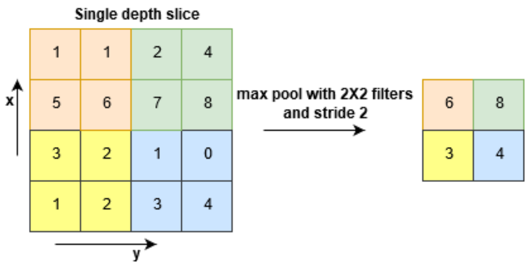

Fig.3: Max pooling with a 2x2 filter and stride 2 [7].

2.2 Training the CNN

Training a machine learning model usually involves choosing an optimization algorithm, defining a cost function and identifying the model family. In deep learning models, the optimization usually takes the form of the backpropagation algorithm. Convolutional neural networks being a subset of the deep learning family of models, follows this same process but with some peculiarities being the addition of the convolution and pooling operations. In general, the training process can be outlined as follows;

1. Image pre-processing: It includes image scaling which resizes all the images in the dataset to the same size
2. Convolution using a kernel: The kernel also known as a feature detector is a 3X3 grid, the kernel is then run over the pre-processed image in strides resulting into a feature map, which is a reduced size of the input as a result of running the kernel over it. Nonlinearity is removed from the resulting feature map using the rectified linear unit (ReLu) function, ReLu trains the neural network more rapidly and without a considerable consequence to generalization accuracy and hence is preferred to other functions such as the hyperbolic tangent and the sigmoid function. Some text presents this step as a separate step from the convolution step, for simplicity the ReLu activation is applied to the feature map as one big step together with the convolution step. The result is a convolved feature map.
3. Pooling: This involves using a window which is a 2X2 grid over the feature map; there are two methods of pooling, the MIN pooling where you take the lowest value on the feature map while running the pool over the kernel and MAX pooling where you take the highest value on the feature map while running the pool over the kernel resulting to a pooled feature map. The pooled feature map goes over the process of convolution and pooling until the performance of the convolutional neural network is maximized, that is, the classification error is minimal.
4. Fully Connected: The resulting pooled feature maps are then fed into the input layer of fully connected neural network as vectors. The significant error is calculated using a cost function then the backpropagation algorithm is used to calculate the gradients of our fully connected network. The value of our gradients must also affect the nature of the convolution and pooling operations in the case of our convolutional neural network and must be adjusted accordingly [8].

2.3 Transfer Learning

Transfer learning is a machine learning technique that deals with improving a new task's learning process by transferring knowledge from an already learned related task. The pretrained model, a model that has been trained to perform certain tasks is used as the initial point for training another model to perform similar tasks. It is used to improve the learning process of a model by transferring knowledge learned from a pre-trained model on a related task thereby saving computational and time resources [9].

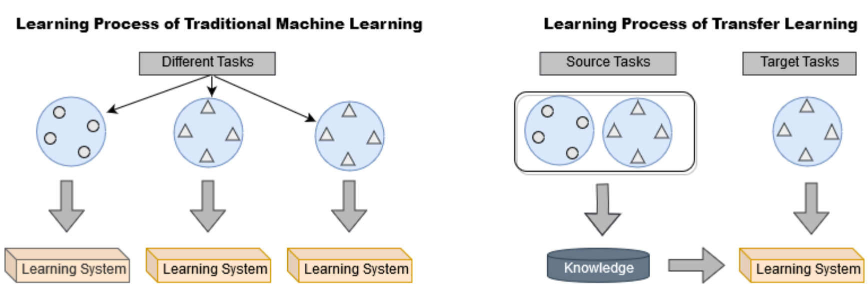

Fig. 4: Learning Process of Traditional Machine Learning Vs Transfer Learning[10].

2.4 Contemporary Nigerian Hairstyles

Hairstyles are an important area in the beauty industry as hairstyles are used to adorn the head. The significance of hairstyles in the Nigerian society is not limited to its beautification aspect as hairstyles have other significant meanings such as culture and events as hairstyles can reflect social status, religious belief, ceremonies, marriage etc.

Traditionally we have three main methods of hairstyling in Nigeria; cornrow, inverted cornrows and threading, the same hairstyle can be portrayed in the different methods. Different ethnic groups have diverse hairstyles as hairstyles were used then to express where a person is from.

Hairstyles have evolved overtime are not just limited to the original hairstyles worn by our fore fathers in the olden days, due to cultural mix and the internet, there has been a crossbreed of hairstyles across the world and the Nigerian scene is not exempted. This has led to hairstyles like bob, pixie cut, setting, perm rods etc. becoming contemporary hairstyles in Nigeria.

Fig. 5: Example of a contemporary Nigerian hairstyle

2.4.1. Cultural Significance of Nigerian Hairstyles

In the African society, hairstyles aren't only meant for the primary purpose of beautification and adornment of the head, they depict a lot of cultural significance and the Nigerian society isn't left out, not only are they used to illustrate where a person is from, they showcase the rich culture of the nation. Certain hairstyles were used to portray royalty as they were only reserved for and only worn by the royals. Some hairstyles were created to portray the political events that occurred in the country, one of such hairstyles is from the Yoruba people of South-Western Nigeria known as "ojukwu dobale fun gowon"; this hairstyle is about the end of the Nigerian civil war and the surrender of the Biafran army, Hairstyles portray one's religious beliefs as different religious had unique hairstyles ascribed to their practices, marital status showcased by the hairstyles worn as there are various hairstyles for brides and married women. Occupation also played a huge role in the types of hairstyles worn as they were used to represent their occupation [1].

2.4.2. Economic Significance

The hair industry has created jobs ranging from the service providers such as the hairstylists and saloon owners to the manufacturers of the products and tools needed for achieving these hairstyles. Thereby reducing the rate of unemployment and boosting the country's economy. The products and tools can also be exported when they meet the required standard.

2.4.3. Information and Communication Technology (ICT) and Nigerian hairstyle industry

ICT can be used to efficiently run the hair business as products and services can be showcased on social media, websites and applications and distributed globally. Building a CNN to identify these hairstyles is relevant for applications in the beauty and fashion industry.

3. Methodology

Given a dataset of 4,805 hairstyle images collected across seven contemporary Nigerian hairstyles which are Cornrows, Shuku, Bob, Afro, Box Braids, Faux Locs and Pixie Cut., the task is to train a neural network to be able to correctly and efficiently classify images across the seven classes of hairstyles and display the results on a visual graphical web interface called hairNet.

3.1 CNN Architecture

The convolutional neural network architecture consists of the convolutional layer architecture and the fully connected classifier and they are explained in the sections below.

3.1.1 Convolutional Layer Architecture

The convolutional layer implements transfer learning, it makes use of a pretrained convolutional neural network VGG-19 that's been trained over ImageNet's database of millions of images. The pretrained parameters of VGG-19 is used as a feature extractor whose output is used to train the fully connected layer of the neural network i.e the classifier. Once the fully connected part of the network is trained, other feature extractors can be used such as the ResNet50 [11], and GoogLeNet-V3 [12]. VGG19 is chosen for its depth[13].

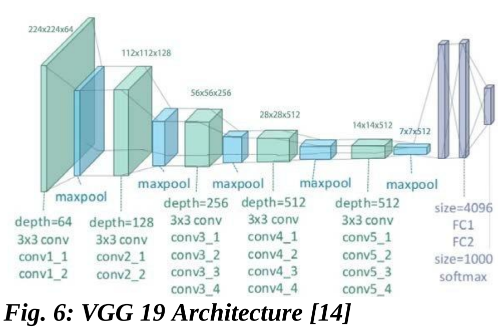

Fig. 6: VGG 19 Architecture [14]

3.1.2 Fully Connected Classifier
The fully connected classifier is a feed forward neural network which performs feed forward and back propagation. As shown in fig. 7 below, the flattened input vector is passed onto the input layer, the input layer has 4096 nodes. After a series of forward pass and back propagation, it produces 29088 nodes which are connected to the hidden. Computation is performed on the hidden layer with its 4096 nodes and produces 4096 nodes connected to the output layer. At the output layer, more computations are performed on its nodes to produce 7 nodes which are the different classes of hairstyles we have.

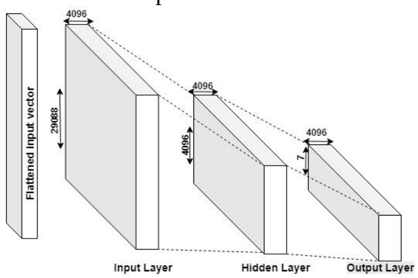

Fig. 7: Fully Connected Classifier

3.2 Model Parameters
The hyperparameters value used in training the convolutional neural network is shown in the Table 1 below. The parameters in table 1 are explained as follows:

Batch Size: A batch size of 64 is used, it represents the number hairstyle images from the training set that propagate through the neural network at each step of the optimizer.

Image Size: The images in the dataset are rescaled to dimensions of 224x224 as required for the VGG19 architecture.

Number of Epochs: The number of epochs refers to the number of iterations the convolutional neural network was trained i.e. its forward step and its backpropagation step. At each iteration, the CNN runs a convolution step for feature extraction and a classification step.

Table 1: CNN model hyperparameters

Batch Size	Image Size	No of Epochs	No of Classifiers
64	64 pixels	12	7

3.3 Optimizer Parameters
The hyperparameters for the Adaptive Momentum (Adam) optimizer follows specification in [15] and[16].
Learning rate (float) – 0.001
Activation function: Rectified Linear Unit (ReLU)
Loss Function: categorical cross-entropy

3.4 Dataset Creation
The Bing web search API was used to scrap the internet for the hairstyle images which were manually filtered and separated to produce a total of 4,805 images across the seven classes of hairstyles and each hairstyle was distributed across the train, test and validate folders in 70:20:10 ratio respectively.

3.5 Data Preprocessing
The dataset contains a total of 4,805 images across the seven hair classes. Two classes are used in the pre-processing; train_transform and test_transform. The train_transfrom defines the transforms for the training set while the test_transform defines the transform for the testing set. The image preprocessing techniques used on both the training and test set are as follows;

Image scaling: Image scaling this is done to ensure uniformity in producing the tensors. PyTorch transform classes are used to resize, normalize, convert pixels to tensor, randomly rotate, crop and horizontally flip the images in the dataset. The Eq.(2) below shows how the resizing is done on the images and its with respect to their initial size.

$$resize = size * \frac{height}{width}, size \qquad (2)$$

Transform to tensor: A tensor is basically a multidimensional array; the pixel values of the images are transformed into a 3D tensor which comprises of the height, width and the RGB channels.

*Normalization:*Normalization was performed on the dataset using the transform. normalize(). Normalization reduces the imbalances by ensuring that every value is within a range and the computational burden on the network by working with smaller numbers therefore preventing the algorithm from placing more priority on larger values. The normalization formula is shown in Eq.(3). below, where x is the input channels and std is the standard deviation.

$$x' = \frac{x - mean(x)}{std(x)} \qquad (3)$$

3.6 Data-flow for Web Application

Fig.8 below is the level 0 diagram for hairNet, the web application. The user can perform the following actions on the web app, classification and querying. Classification is performed by the user uploading a hairstyle and the application returns the hairstyles that is predicted as well as the other information pertaining to the classified hairstyle. The User can also search directly for information about hairstyles and the result. The administrator only maintains the database.

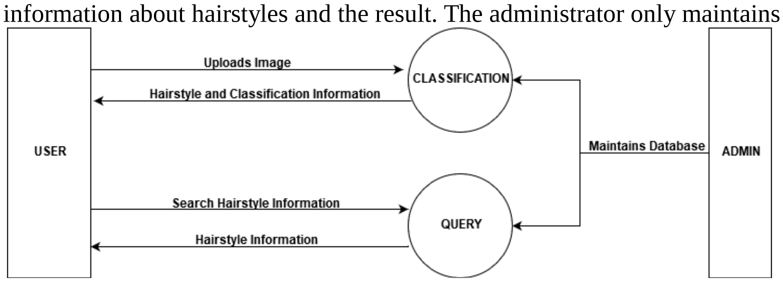

Fig.8: hairNet Contextual Diagram - Level 0 DFD

3.7 Use Case for Web Application

There are two actors in the web application, the user and the admin. The administrator performs system maintenance to ensure smooth user experience while the user can perform the following tasks:

Hairstyle Classification: where the image of the hairstyle is uploaded to hairNet, this image is then tested against our model and then the hairstyle is classified and as displayed along with other information related to the classified hairstyle.

Search: The user can also query hairNet for information pertaining hairstyles. The use case diagram for the web application is presented in fig.9:

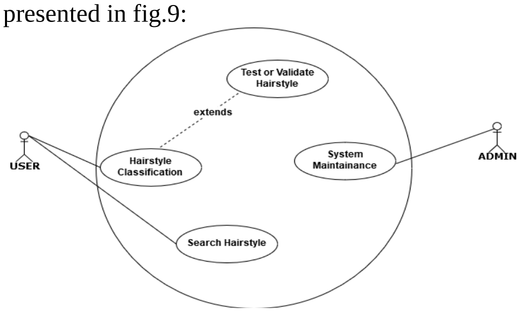

Fig. 9: Use-Case Diagram for hairnet

3.8 Experimental Setup

The source code for the convolutional neural network is coded in python and it makes use of the PyTorch deep learning library, the training is implemented on Google Colab. The coding tools used to develop the web application hairNet includes: Hypertext Markup Language (HTML), Cascading Style Sheets (CSS) Javascript and Django.

4.0 Evaluation and Results

4.1 Training Result:

The convolutional neural network was trained on the 7 classes of hairstyles at 12 epochs and training time for each epoch was 2 hours 15 minutes. The network couldn't be trained longer because it was trained on one GPU. Fig. 10 shows the training result for each epoch, which includes, the training loss and the validation accuracy.

```
Epoch: 1        Training Loss: 3.407234         Validation Loss: 1.715106
Valid Accuracy: 48.0000 %
Validation loss decreased (inf --> 1.715106).  Saving model ...
Epoch: 2        Training Loss: 2.212252         Validation Loss: 1.070890
Valid Accuracy: 58.0000 %
Validation loss decreased (1.715106 --> 1.070890).  Saving model ...
Epoch: 3        Training Loss: 2.000830         Validation Loss: 1.016525
Valid Accuracy: 74.0000 %
Validation loss decreased (1.070890 --> 1.016525).  Saving model ...
Epoch: 4        Training Loss: 1.955448         Validation Loss: 0.858177
Valid Accuracy: 74.0000 %
Validation loss decreased (1.016525 --> 0.858177).  Saving model ...
Epoch: 5        Training Loss: 1.911389         Validation Loss: 0.802722
Valid Accuracy: 76.0000 %
Validation loss decreased (0.858177 --> 0.802722).  Saving model ...
Epoch: 6        Training Loss: 1.896747         Validation Loss: 0.856379
Valid Accuracy: 62.0000 %
Epoch: 7        Training Loss: 1.812151         Validation Loss: 0.725493
Valid Accuracy: 84.0000 %
Validation loss decreased (0.802722 --> 0.725493).  Saving model ...
Epoch: 8        Training Loss: 1.860076         Validation Loss: 0.812086
Valid Accuracy: 82.0000 %
Epoch: 9        Training Loss: 1.760977         Validation Loss: 0.718234
Valid Accuracy: 86.0000 %
Validation loss decreased (0.725493 --> 0.718234).  Saving model ...
Epoch: 10       Training Loss: 1.763576         Validation Loss: 0.726720
Valid Accuracy: 80.0000 %
Epoch: 11       Training Loss: 1.672158         Validation Loss: 0.690353
Valid Accuracy: 78.0000 %
Validation loss decreased (0.718234 --> 0.690353).  Saving model ...
Epoch: 12       Training Loss: 1.709448         Validation Loss: 0.706787
Valid Accuracy: 80.0000 %
```

Fig. 10: Training Result

In fig. 11, one can observe that the training loss reduced steadily, showing that the model was improving over each epoch. Yet, the training accuracy remained within the range of 80-83%.

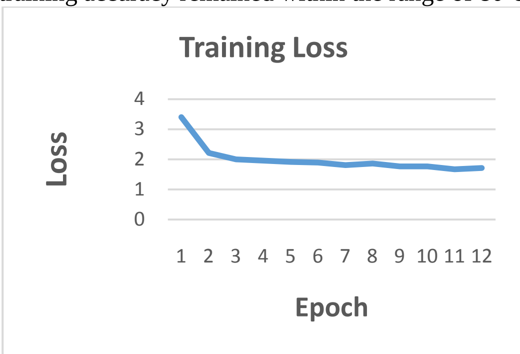

Fig. 11: Training

4.2 Validation Result

Images in the validation folder was used to validate the result gotten from training the network to further improve the feature extraction. Fig. 12 shows the validation loss for the model. The validation loss performance is slightly better than the training loss over the epochs. This is a good indication that the model is not overfitting to the training data. The validation loss reduces steadily down to 0.7, but since the validation accuracy itself was not increasing by the 10th epoch, training was stopped to avoid overfitting.

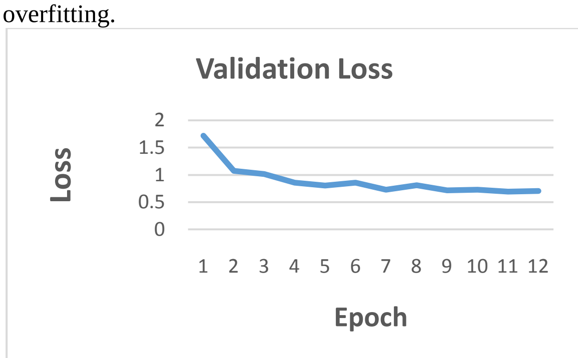

Fig. 12: Validation Loss

4.3 Test Result

After training and validation, the result is then tested against the training set with an average test accuracy of 86%. Fig. 13shows the test result against a sample image as well as the class probability

Fig. 13: Test Result Against One Image

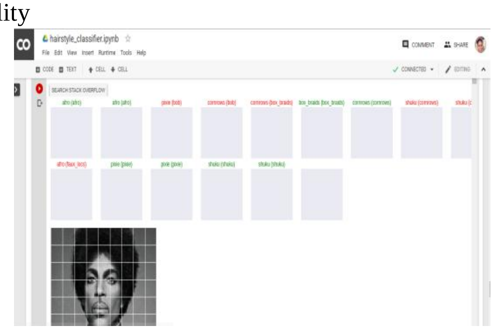

Fig. 14: Test Result Against a Set of Test Images

Fig. 14 shows the test result across a group of test images showing the ones that was correctly classified and the ones that was wrongly classified.

`

4.4 HairNet Application Interface

HairNet is a dynamic web app, it accepts input from the user in form of hairstyle images and returns a classification for the image as well as the probability for the hairstyle in each of the classes, also containing information about the hairstyle class of the uploaded image. Once the image is uploaded, the classification is done by loading the convolutional neural network model embedded in the app, the result is then displayed for the user. The homepage is displayed when the app is launched, it contains a “predict hairstyle” button which is clicked on by the user to upload images and predict the hairstyles from the available classes. Fig. 15 shows the hairNet homepage and it contains the predict hairstyle button which directs the user to the image upload section of the page.

Fig. 15: Homepage

Fig.16 shows the upload section where the images to the classified are uploaded. It contains a “add image” button that’s used to upload an image from the local computer and a drag and classified.
drop area where images can be dragged and dropped for classification.

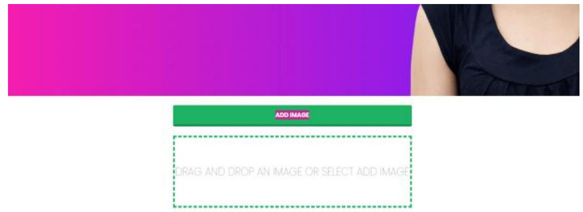

Fig.16 Image Upload

Fig. 17 Shows the process of uploading the image once the “add image” button is clicked. A file explorer is opened, the user can then navigate amongst all the files and folder on the computer and select the image that is to be

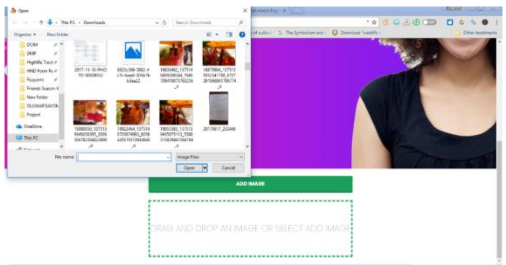

Fig. 17 Image Upload Process

Once a picture is selected, it shows a preview of the selected image, there are also two buttons, the red button gives the user the option of changing or removing the image if the wrong image was selected while the blue button is used to classify the selected image, this can be seen in Fig. 18.

Fig. 18 Selected Image

Once the user clicks on the identify button a loading, the image is then directed to the backend for identification and while this is going on, a gif is displayed on the screen till the result has been computed and is ready to be displayed and this can be seen in fig. 19.

Fig.19 Loading

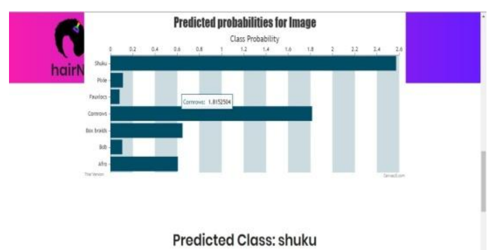

Fig. 20: The predicted class along with the predicted probability range

Once the prediction is done, the classification result is displayed. Fig. 20 shows the predicted class of the selected image which is “shuku” alongside the predicted probability for the image in a bar chart and shows the predicted probability for each hairstyle class.

Information about the predicted hairstyle class is also displayed such as a description of the hair, how to make it as well as saloons nearby where the hairstyle can be made, this can be seen in fig. 21.

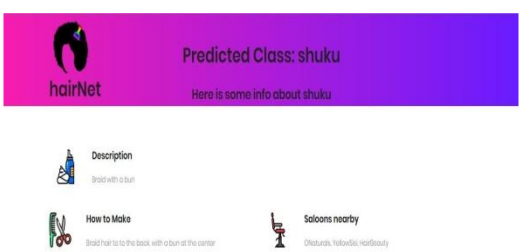

Fig. 21: Information on the predicted hairstyle class

5. Conclusion

The project successfully trained and tested a convolutional neural network on a novel Nigerian hairstyle image dataset. The VGG19 model used for transfer learning demonstrates the power of transfer learning for the image classification task. The model achieved high

validation and test performance with test accuracy of 86%. This was achieved with a relatively small dataset of 4850 instances, there is room for improvement with training on a larger dataset. The convolutional neural network is also successfully deployed as a light weight web application, therefore demonstrating the potential of CNN based web solutions across a wide range of tasks. There is room for future work that improves upon the scope and limitations of this work. One of the paths for future studies using convolutional neural networks in image classification include using other loss functions for training the convolutional neural network.

References

[1] Houlberg, M.H., 1979. Social Hair: Tradition and Change in Yoruba Hairstyles in Southwestern. The fabrics of culture: The anthropology of clothing and adornment, p.349.

[2] Goodfellow, I., Bengio, Y., Courville, A. and Bengio, Y., 2016. Deep learning (Vol. 1). Cambridge: MIT press.

[3] Albelwi, S. and Mahmood, A., 2017. A framework for designing the architectures of deep convolutional neural networks. Entropy, 19(6), p.242.

[4] Bengio, Y., 2009. Learning deep architectures for AI. Foundations and trends® in Machine Learning, 2(1), pp.1-127.

[5] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[6] Gregor, K. and LeCun, Y. (2010). Emergence of complex-like cells in a temporal product network with local receptive fields. Technical report, arXiv:1006.0448.

[7] Sudhakar, S. (2017). Convolution Neural Network – Towards Data Science. [online] Towards Data Science. Available at: https://towardsdatascience.com/convolution-neural-network-e9b864ac1e6c [Accessed 18 Jul. 2018].

[8] Kim, Y., 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

[9] Torrey, L. and Shavlik, J., 2010. Transfer learning. In Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques (pp. 242-264). IGI Global.

[10] WenyuanDai, QiangYang, Gui-RongXue, and YongYu. Boosting for transfer learning. In Proceedings of the 24th international conference on Machine learning, pages 193–200. ACM, 2007.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings indeep residual networks. arXiv preprint arXiv:1603.05027,2016.

[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed,D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich.Going deeper with convolutions. In Proceedings of the IEEEConference on Computer Vision and Pattern Recognition,pages 1–9, 2015.

[13] Saikia, A.R., Bora, K., Mahanta, L.B. and Das, A.K., 2019. Comparative assessment of CNN architectures for classification of breast FNAC images. Tissue and Cell, 57, pp.8-14.

[14] Zheng, Y., Yang, C. and Merkulov, A., 2018, May. Breast cancer screening using convolutional neural network and follow-up digital mammography. In Computational Imaging III (Vol. 10669, p. 1066905). International Society for Optics and Photonics.

[15] Radford, A., Metz, L. and Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

[16] Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.