Animal Classification using Facial Images with Score-Level Fusion

Y. M. S. D. Sastry, Assoc. Prof. CSE:BVCE, ymsdsastry.bvce@bvcgroup.in
B. N. Rao, Assoc. Prof. CSE:BVCE, bnraolak@gmail.com
Manepalli Durga Prasad, CSE:BVCE
Mattaparthi Nandini Bhanusri, CSE:BVCE
Ravuri Devi Gayatri Anusha, CSE:BVCE
Saladi Radha Krishna, CSE:BVCE
Peddireddy Valli Baba, CSE:BVC

Article Info
Page Number: 5320 - 5337
Publication Issue:
Vol 71 No. 4 (2022)

Article History
***Article Received**: 25 March 2022*
***Revised**: 30 April 2022*
***Accepted**: 15 June 2022*
***Publication**: 19 August 2022*

Abstract

A new area of study in machine vision is a real-world animal biometric system that can find and describe animal life in image and video data. These systems use computer vision to figure out how to put animals into groups. We show a new way to classify animal faces based on the score-level combination of recently popular convolutional neural network (CNN) features and appearance-based descriptor features. This method uses a score-level fusion of two different approaches. One uses CNN, which can automatically extract, learn, and classify features, and the other uses kernel Fisher analysis (KFA) to extract features. The proposed method could also be used to classify images and recognise objects in other ways. The results of the experiments show that automatic feature extraction in CNN is better than other simple feature extraction techniques, both for local and appearance-based features. Also, a combination of CNN and simple features with the right score level can get even better results than using CNN alone. The authors showed that the score-level fusion of CNN-extracted features and the appearance-based KFA method have a positive effect on classification.

accuracy. The proposed method can classify animal faces with a rate of 95.31 percent, which is much higher than the current best methods.
CNN, kernel Fisher analysis (KFA), HOG, median robust extended local binary pattern (MRELBP), and biologically inspired features are some of the words that describe them (BIFs).

I. INTRODUCTION

Identifying and classifying animals is an important topic that hasn't been talked about much. Animal classification that is based on being able to tell the difference between images of different species of animals is easy for humans to do, but there is evidence that even in simple cases like cats and dogs, it is hard for computers to tell them apart. Animals have flexible bodies that can change their own looks, and they usually show up in complex scenes. Also, like all things, they can look different depending on the light, the angle, and the size. There are attempts to use recognition methods on images of animals, but the problem of putting animals into groups hasn't gotten much attention lately.

Many of the methods that are already available and show promise for human face recognition can't properly represent the diversity of animal classes with their complex intra-class differences and inter-class similarities. There are many ways to solve this problem, and each has pros and cons.

• The first method builds complex features that better represent and separate sample images, but making these features is hard and depends on the problem.

• In the second method, extracted features from different methods are added together to make a stronger feature vector. When the size of the feature space grows, the cost of solving the problem goes up. Instead of making a complicated representation, information from different classifiers is put together and used to make a decision.

1.1 Score-Level Fusion:
It works very well in biometric systems that use more than one type of sensor. In a biometric recognition system, the match score is a way to figure out how similar the biometric feature vectors from the input and the template are. When the match scores from different biometric matchers are added together to make a final recognition decision, this is called score-level fusion or fusion at the match score level. Aside from the raw data and feature vectors, the match scores have the most information about the input pattern. Also, it's not too hard to get to the scores made by different biometric matchers and add them together. Many research studies have used score-level fusion. In, the authors suggested using a score-level combination of fingerprint and face matchers to verify a person's identity when they are under stress. The system used a score-level combination of the person's fingerprint and voice to identify them. It showed a way to combine face and iris biometric traits with the weighted score-level fusion technique to combine the matching scores from these two modalities based on their weight availability. Using the weighted fusion technique, a person's fingerprint, palm print, and iris are all added together in a multi-biometric system.

The term "multi-biometric system" refers to the use of more than one source of biometric information to figure out who someone is.

By combining two or more biometric modalities, a multi-modal biometric system can cause problems. Compared to biometric recognition systems that only use one type of biometric, multimodal biometrics give a lot more information. For multimodal biometric systems, you need a fusion framework and a good recognition algorithm.

1.3 Methods for Feature Extraction: An important part of a biometric system is the step of Feature Extraction. How features are extracted has a big impact on how a biometric recognition system makes its final choice. In this study, a number of feature extractors that have been used in the past are put into place. These feature extraction methods are called convolutional neural network (CNN), histograms of oriented gradients (HOGs), median robust extended local binary pattern (MRELBP), kernel Fisher analysis (KFA), and biologically inspired features (BIFs).
Some feature extraction methods are mentioned below

HOG, which stands for histograms of oriented gradients, is a popular feature extractor that has been used to classify objects and recognise faces. This method builds the image's features by counting the number of times a gradient orientation shows up in local patches or the detection window. In this method, an image is broken up into parts, and the histogram of orientations for each part is then calculated. The HOG feature is made by putting all of these histograms together. HOG is used in several recognition systems.

Median robust extended LBP: The Local Binary Patterns (LBPs) method is thought to be one of the best high-performance texture descriptors in terms of how well it works with computers. LBP is used in several recognition problems such as face, iris, palmprint and plant recognition. But the LBP method is very sensitive to image noise and can't pick up information about the macrostructure. In order to best address these disadvantages, a novel descriptor for texture classification, namely MRELBP. MRELBP doesn't use raw image intensities; instead, it compares the median of image intensities in a local area. This method can get information about both the texture of the microstructure and the texture of the macrostructure. It does this by using a multiscale LBP-type descriptor and a new sampling scheme. Salt-and-pepper noise, Gaussian noise, Gaussian blur, and random pixel corruption have all been shown to have very little effect on MRELBP. When compared with large number of LBP variants. They set up different tests to see how well their feature descriptors handled changes in rotation, view point, illumination, scale, different types of image degradation, number of classes, and computational complexity.

Biologically inspired features

BIFs are a way to try to model how the cortex processes visual information as a stack of layers that get more complex as they go up. The "HMAX" model is a new set of features that come from a feed-forward network of the visual object recognition pathway in primates. The model is made up of two types of layers: S units (simple neurons) and C units (complex neurons) (C). One interesting thing about this model is that it uses the non-linear maximum operation (MAX) in the neurons of the simple units instead of the linear summation operation (SUM) when pooling inputs at the layer called C1. The maximum values within local patches and across scales within a band are calculated. So, the C1 feature has eight bands and four ways to use them. The bio-inspired features (BIFs) have been looked into for face recognition and recognising the type of an object.

1.4 Motivation

The goal of this study is to find out how deep learning algorithms can help better classify animals based on their faces and how combining these models can make them work better than the ones that are already out there, since there isn't a single paper that talks about the classifications done in this. Lastly, to know and understand how these models can be used to classify animals based on their faces and how they differ from each other.

1.5 Problem Statement: This is a problem because it is hard to put animals into groups. It is hard to figure out how they evolved together and what parts they share. The cost of computation also goes up when the size of the feature space grows.

1.6 Goals: Our main goal is to make a system that will use computer vision techniques to sort animals into groups.
1.7 The work to be done:
This project is about:

To put animals into groups based on their faces using score level fusion.
2. Here, it's easy to get to the scores made by different biometric matchers and add them together.
3.In this process, we use multiple sources of biometric information to figure out who someone is.

2: With the help of deep learning algorithms, research has been done on how to classify animals based on their faces. CNN and KFA haven't been used much to figure out how to classify things. This research helps improve the way animals are put into groups. The two deep learning models give us a way to calculate the confusion matrix graph for our problem statement, which is driven by nature. This research helps a lot with putting animals into groups based on their faces.

II. System Planned
Based on how well score-level fusion works in multimodal biometric recognition systems, it is thought that the information from two different types of classifiers can be combined to improve accuracy. Because there is a lot of similarity between classes and a lot of difference within classes, we need to combine two different kinds of feature descriptors. When CNN and other simple features are used together at the right score level, they can be even more accurate than CNN alone. In our proposed method, we combined features that were taken from two different descriptors at the score level.

In this paper, the author is taking features from two different classifiers, such as KFA (Kernel Fisher Analysis) and Convolution Neural Networks (CNN), to improve classification of animal facial images. In this proposal, two separate algorithms will be trained on the "LHI-Animal-Faces Dataset," then features will be extracted and a score calculation function will be used to pick the class label from the classifier with the highest score. It's called a "Fusion Score" when the results of both classifiers are added together to predict a class label.

3.1.1 Convolutional Neural Network
CNN is a powerful deep learning machine learning technique that has been used in many computer vision tasks [14]. In this study, we use a CNN that has already been trained as a feature extractor to find different ways to represent animal faces. In order to train a CNN, you need a very large set of training images. CNNs automatically learn to extract discrimination features from these large training set images. In most cases, these features are better than hand-crafted features like HOG, local binary pattern (LBP), or speeded-up robust features (SURF).

Traditional CNNs use the soft max multi-category classifier with the cross-entropy loss function and have a stack of convolutional layers followed by some fully connected layers. First, we'll briefly talk about each of CNN's layers:
In general, a Neural Network has three layers:

1. The Input Layer
2. Hidden Layer (can consist of one or more such layers)
3. Layer for Output

The Hidden layer can be broken down into three main layers.

1. Convolutional layer: The goal of this layer is to find features that don't change based on where they are in space by using different convolutional filters.
2. Pooling layer: To get rid of high-frequency noises, the extracted feature maps of the convolutional layer are down-sampled by taking the local maximum (max-pooling) or average (avg-pooling) value of each patch in the feature map.
3. Fully connected layer: Each neuron in this layer is connected to all of the neurons in the two layers on either side of it.

Usually, there are two steps in a convolutional neural network process.

Forward Propagation: The weights and biases are set by chance at the beginning, and this is how the output is made at the end.
Back Propagation: At the start, the weights and biases are set by chance, and the values are changed based on the error. Forward propagation is done over and over again with these updated values for newer outputs to reduce error.

In the next sections, we'll talk briefly about AlexNet and VGG-16, two of the most popular CNN architectures that we used to automatically pull out features from animal faces.
AlexNet: It is a deep CNN for classifying images that won the ImageNet large-scale visual recognition challenge (ILSVRC)-2012 competition with a top-5 test error rate of 15.3%, while the second-best entry had a top-5 test error rate of 26.2%. Alexnet has eight layers with parameters that can be learned. It is made up of five convolutional layers (C1–C5), two fully connected layers (FC6 and FC7), and a softmax output layer as the last layer (FC8). The only layer that does not use Relu activation is the output layer. All together, there are 62.3 million parameters in this architecture. The diagrams below show how AlexNet is put together.

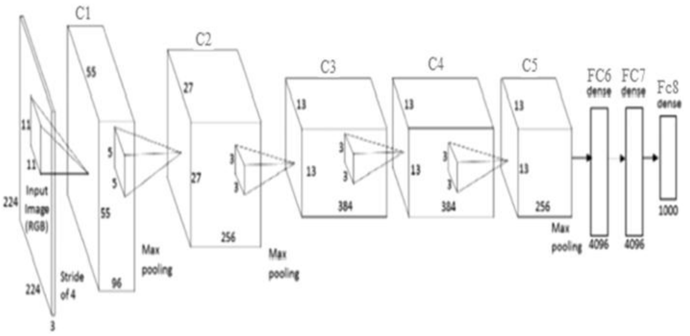

Fig.1. AlexNet architecture

VGG-16*:* This architecture had been proposed in ILSVRC 2014. The Oxford Visual Geometry Groups' model is deeper and wider than former CNN structure. VGG-16 has five batches of convolution operations, each batch consist of 3–5 adjacent convolution layers. Adjacent convolution batches are connected via max-pooling layers.The size of kernels in all convolutional layers is 3×3 convolutional layers and the number of kernels within each batches is the same (increases from 64 in the first group to 512 in the last one). Bellow Fig illustrates a 16-layer VGG architecture. VGG-16 network architecture has been used in many researches and it was the first one that outperformed human-level performance on ImageNet.

The 16 layers of VGG16
1.Convolution using 64 filters
2.Convolution using 64 filters + Max pooling
3.Convolution using 128 filters
4. Convolution using 128 filters + Max pooling
5. Convolution using 256 filters
6. Convolution using 256 filters
7. Convolution using 256 filters + Max pooling
8. Convolution using 512 filters
9. Convolution using 512 filters
10. Convolution using 512 filters+Max pooling
11. Convolution using 512 filters
12. Convolution using 512 filters
13. Convolution using 512 filters+Max pooling
14. Fully connected with 4096 nodes
15. Fully connected with 4096 nodes
16. Output layer with Softmax activation with 1000 nodes.

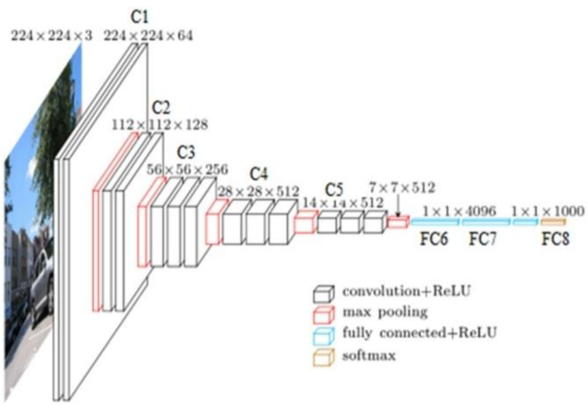

Fig.2. VGG16 architecture

3.1.2 Kernel Fisher discriminant analysis:

This KFA method goes further than Fisher's discriminant analysis (FDA). In this method, the first step is to use a nonlinear mapping to make the input space bigger. The next step is to apply the multiclass FDA to the larger feature space. By using non-linear mapping, the number of dimensions in the feature space will grow, which makes the KFA method better at telling things apart. The main benefit of the KFA method is that it can be used to classify patterns that belong to more than one group. Its solution is also unique, which makes it better than the generalised discriminant analysis [25] method, which gives more than one answer.

Advantages of Proposed System: In the proposed paper, the author used all deep learning algorithms, such as VGG, RESNET, with and without KFA features. We also used an advanced XGBOOST algorithm with KFA, which can train data accurately because it has features like an optimised distributed gradient boosting library that is designed to be highly efficient, flexible, and portable. We're making KFA features even better by using the PCA algorithm. This improvement, along with XGBOOST, brings the accuracy up to 100%.

3.3 ARCHITECTURE/Framework:

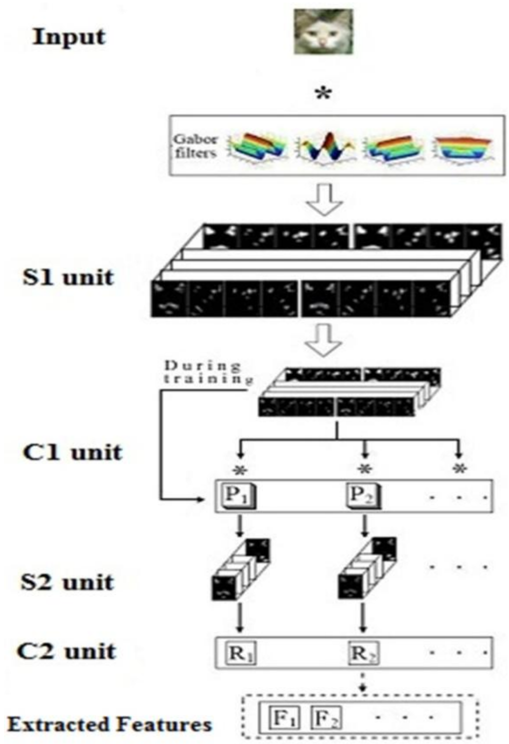

FIG3. Architecture

3.4 Algorithm and Process Design:

This is explained in more detail below. In the pre-processing step, we do simple things to each animal head image that comes in, like resize it, change it from RGB to grayscale, and even out the histogram, to get rid of the negative effects of things like size, lighting, and picture quality. But we only change the size of the image that comes in for CNN features. Then, two different sets of features are calculated. Then, we compare how similar these feature vectors are to all of the feature vectors in the training set and choose the one with the least similarity for each method. The score for that sample is thought to be the distance between the images of the test animal and the images of the training animal. In the next step, we took these scores and made sure they were all the same. In the end, we make a decision by using a classifier called "nearest neighbour" (NN) that uses "normalised fused scores."

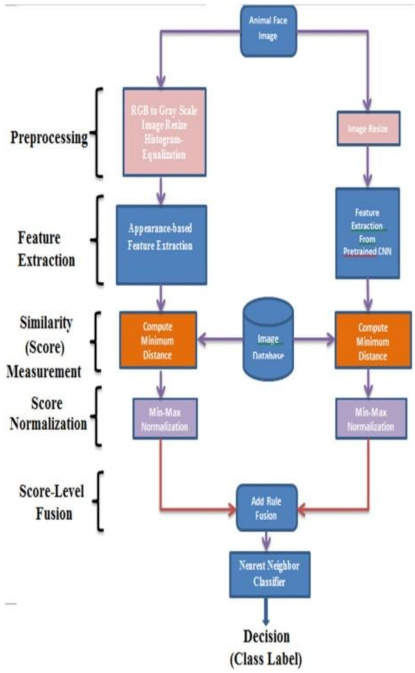

Fig.4. Preprocessing

Pre-processing:

In this module images will be converted to grey colour and then resize images and then apply histogram calculation to normalise images data

Features Extraction: All process images will be given to the KFA algorithm, which will then pull out the important parts of each image and make a list of important parts.

Similarity score: The features of the input images and the features of the KFA will be compared to see how similar they are, and the feature with the smallest difference will be used for training.

Normalize score:

In score will be normalise by applying division or average function Score level fusion: This score will be added to the CNN algorithm, and then the next-door neighbour method will be used to predict the class label with a high score.

Add Rule Fusion:

In this module rules that determine the exact decomposition of the tensor product of two representations of a group into a direct sum of irreducible representations. IV.

IMPLEMENTATION AND OUTCOMES

Several tests have been done to compare how well the proposed method works with other methods that are currently in use. In the sections that follow, we talk about the dataset, how the experiment was set up, and what the results were. In order to eliminate the negative effect of different factors such as size, illumination and picture quality, all the images have been resized to 60×60 pixels and perform pixel intensity normalisation.

In order to get features out of an image, the image is shrunk and fed to the CNN as a set of intensities for each pixel. The size of the image that goes into the VGG-16 net is a 2242243 matrix, while the size of the image that goes into the AlexNet architecture is a 2272273 matrix. Table 1 shows that the accuracy of the classifier trained with AlexNet features is close to 90%, and the accuracy of the classifier trained with VGG-16 features is close to 93%. Both of these are higher than the accuracy of the classifier trained with hand-crafted features like LBP and HOG. AlexNet isn't as good as VGG-16 because AlexNet isn't as deep as VGG-16. VGG-16 has 16 layers, 13 of which are convolutional and 3 of which are fully connected. VGG-16 uses small 3-by-3-pixel convolutional filters, so each filter can pick up simpler geometric structures. However, the increased depth of VGG-16 lets it make more complex decisions.

4.1 About the Data: The LHI-Animal-Faces dataset is made up of 2200 images from 19 classes of animal heads and one class of human heads. Fig. 5 shows five sample images from each of these categories. Unlike other general classification datasets, LHI-Animal-Faces only has animal or human faces. These faces have a lot of similarities within their own classes (because of evolution, some animal face categories are similar to the other class) and a lot of differences between classes (see Figure 6). (rotation, posture variation, subtypes). We use 30 images of each type of animal for training, and the rest of the images in each type are used for testing.

4.2 Evaluation Metrics:

We made confusion matrices for both the case of fine-tuned VGG-16 alone and the case of score-level fusion of VGG-16 and KFA to show how well each class was classified. For seven classes, the accuracy of the classification is 100%, and for a lot of other classes, it is good enough and close to 100%. In the fine-tuned VGG-16 confusion matrix, deer head and rabbit head versus dog head (12%) cause the most confusion. The score-level fusion of fine-tuned VGG-16 and KFA brings these confusion values down to 0 and 4%, respectively. In the proposed method, bear head and pigeon head and rabbit head and mouse head cause the most confusion (8%). In the fine-tuned VGG-16, 11 classes are confused with the dog head class. In the Score-level fusion of the fine-tuned VGG-16 and the KFA, this number dropped to four classes.

Accuracy: It's the most important way to measure how well something works, and it's easy to do with a ratio of the number of correct predictions to the total number of observations.

Accuracy = TN plus TP/TN plus TP plus FN plus FP

The confusion network is a way to measure how the show is being judged.

For this, values are calculated based on: • True positive (TP) = number of events for which the correct answer was given. • False negative (FN): The number of events that were wrongly predicted and did not happen. • False-positive (FP) = the number of wrongly predicted events. • True negative (TN): The number of events that could have happened but didn't.

Table 1 Classification accuracy of different methods on LHI-Animal-Faces dataset

Type of methods	**Method**	**Accuracy,%**
local feature descriptor methods	HOG	66.54
	LBP	61.74
	CLBP [33]	63.59

	Fourier-LBP [34]	50.29
	Haralick feature	49.27
	BIF	68.46
	median robust CLBP (MRCLBP)	68.46
appearance-based feature descriptor methods	LDA	60.33
	KFA	69.87
CNN features	FC7 AlexNet features	89.91
	FC7 VGG-16 features	92.84
	Fine-tuned AlexNet	91.06
	Fine-tuned VGG-16	94.39
score-level fusion methods	LDA + HOG	74.32
	LDA + LBP	68.91
	LDA + CLBP	70.23
	LDA + Fourier-LBP	62.44
	LDA + Haralick feature	61.59
	LDA + BIF	77.26
	LDA + MRCLBP	76.30
	LDA + FC7 AlexNet features	90.61
	LDA + FC7 VGG-16 features	93.77
	KFA + HOG	76.48
	KFA + LBP	74.19
	KFA + CLBP	74.14
	KFA + Fourier-LBP	72.65
	KFA+ Haralickfeature	70.94
	KFA + MRCLBP	78.98
	KFA + FC7 AlexNet features	91.37
	KFA + FC7 VGG-16 features	94.21
	KFA + FC7 fine-tuned AlexNet features	92.86
proposedmethod (KFA+ FC7fine-tuned VGG-16 features)		95.31

We tried out different score-level fusions of these features, and some of the best ones are listed here.

The distance between each test sample and its closest training sample is thought to be the score of that test sample in the corresponding classifier.

The min-max method of normalisation: It is a common way to make data consistent. For each feature, the minimum value becomes a 0, the maximum value becomes a 1, and every other value becomes a decimal between 0 and 1.

The min–max normalisation method is used to make these scores equal:

$x' = x - \min(x)/\max(x) - \min(x)$

Where x is the raw score, max x is the highest raw score, min x is the lowest raw score, and x′ is the normalised score.

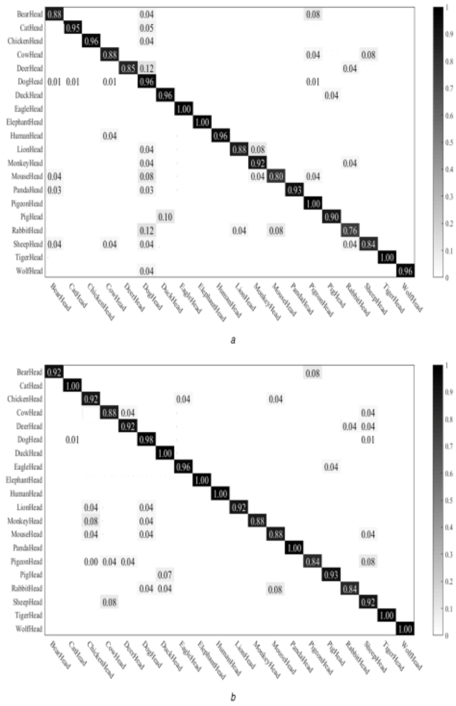

Fig.7. Confusion matrix

a) VGG-16 was tweaked

b) Score-level fusion of finely tuned VGG-16 and KFA

After score normalisation, the multimodal score vector x1, x2 can be made, with x1 and x2 being the normalised scores of two different systems. The next step is to fuse at the level where the scores match. The sum rule-based fusion method is used to combine the scores into a single scalar score, which is then used to make the final decision.

$f_s = w_1x_1+w_2x_2$

The symbol wi stands for the weight that is given to one of the two systems. This weight shows how important each system is in comparison to the other. In Table 1, w1 is the weight of the first method in the (method1 + method2) syntax. In all of our experiments with the score-level fusion method, we used the grid-search algorithm to find the best value for w1 and w2 by giving them different values between 0 and 1 and w1 = 1 w2. The best values for the proposed method are w1=0.3 and w2=0.7. This shows how much more important CNN is than KFA.

The results of the experiments show that score-level fusion always makes accuracy better in a meaningful way. Table 1 shows the accuracy of different feature descriptor methods on the LHI-Animal-Faces dataset when used alone and when their scores are combined. The table shows that the proposed method, which uses score-level fusion with the FC7 activation feature of fine-tuned, pre-trained VGG-16 and KFA, has a classification rate of 95.31 percent, which is higher than all the other methods in the table. So, it can be said that the proposed method is better than all the other

ISSN: 2094-0343
2326-9865

local and appearance-based methods and all the possible combinations of these methods with score-level fusion.

Table 2 Classification accuracy on LHI-Animal-Faces dataset

Method	Accuracy, %
HOG+SVM[13]	70.8
HIT [13]	75.6
LSVM [38]	77.6
AOT [32]	79.1
deep boosting [11]	81.5
proposed method	95.31

4.3 Expected Outcome

The Summary of validation accuracy and min-max normalisation method metrics of one deep learning model i.e, CNN was employed as predictors will be illustrated based on the tested results of our proposed model which performs better in classification of animals using facial images.

Fig.8. Upload LHI-Animal-Faces Dataset'

In above diagram click on 'Upload LHI-Animal-Faces Dataset' button to upload dataset and to get below diagram

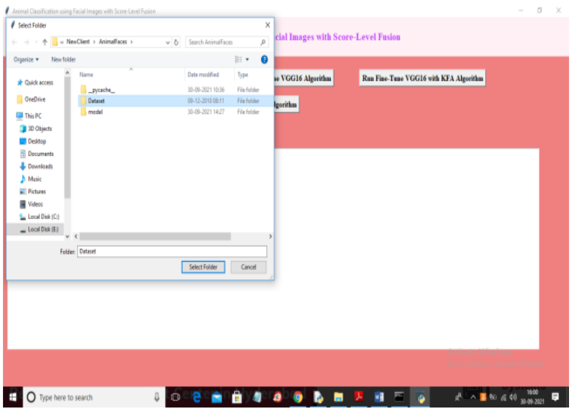

Fig.9. and uploading entire 'Dataset'

In above diagram selecting and uploading entire 'Dataset' folder and then click on 'Select Folder' button to load dataset and to get below diagram

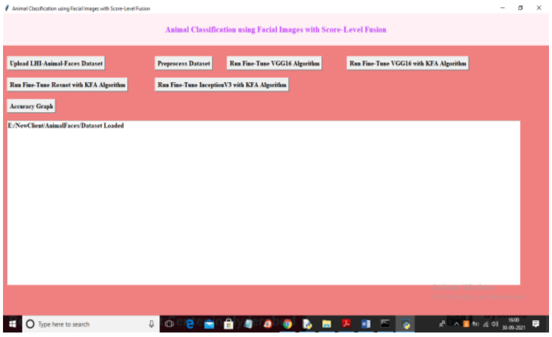

Fig.10. dataset loaded

In above diagram dataset loaded and now click on 'Preprocess Dataset' button to process dataset and to get below diagram

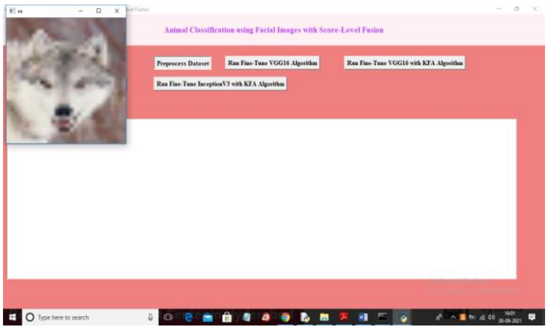

Fig.11. images are processed

In above diagram all images are processed and for sample I am displaying one image to check whether image process correctly or not and now close above image to get below diagram

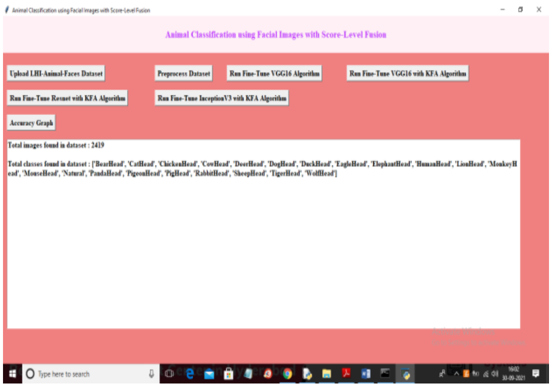

Fig.12. Images in dataset

In above diagram we can see dataset contains total 2419 images and displaying labels of all images available in dataset and now click on 'Run Fine-Tune VGG16 Algorithm' button to run existing VGG16 algorithm and to get below output

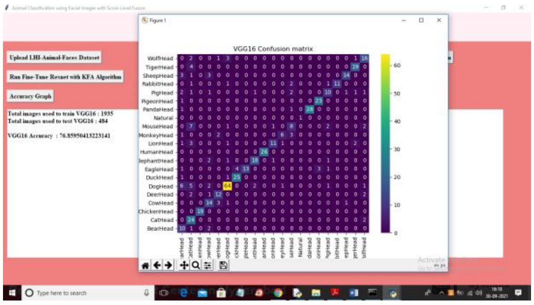

Fig.13. VGG16 got 76% accuracy

In above diagram existing VGG16 got 76% accuracy and in confusion matrix graph all the values in diagnol boxes will be consider as correct prediction and the remaining values are incorrect prediction. In above graph we can see so many values are there in outside of diagnol boxes so plain VGG16 prediction is not accurate and now click on ‘Run Fine-Tune VGG16 with KFA Algorithm’ button to combine features from KFA and VGG16 and then predict with highest score to get below output

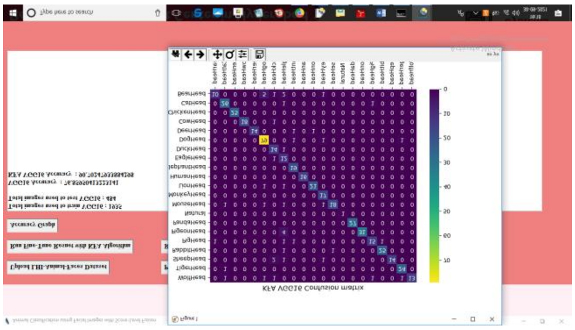

Fig.14. KFA-VGG16 we got 90% accuracy

In above diagram with KFA-VGG16 we got 90% accuracy and in above graph we can see very few numbers are there outside of diagnol boxes as incorrect prediction and VGG16 with KFA can be consider as accurate in prediction and now close above graph and then click on ‘Run Fine-Tune Resnet with KFA Algorithm’ button to run extension RESNET algorithm and to get below output

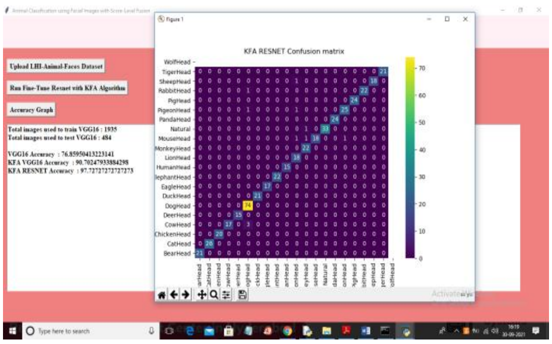

Fig.15. Extension KFA-Resnet we got 97% accuracy

In above diagram with Extension KFA-Resnet we got 97% accuracy and now click ‘Run Fine-Tune InceptionV3 with KFA Algorithm’ button to combine KFA and InceptionV3 features to get fusion prediction

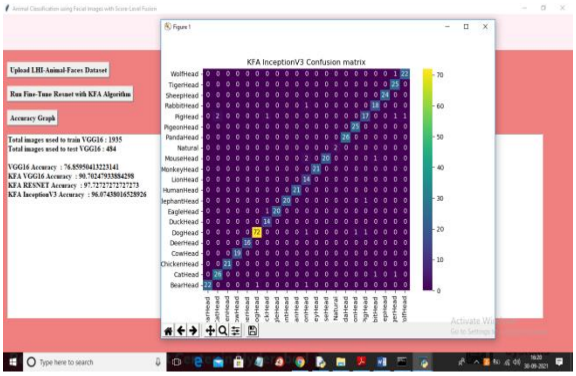

Fig.16.with InceptionV3 we got 96% accuracy

In above diagram with InceptionV3 we got 96% accuracy which is higher than propose KFA-VGG16 accuracy and in above confusion matrix we can see there are last number of wrong prediction outside of diagnol boxes compare to existing VGG16 and propose KFA-VGG16 and now close above graph and then click on ‘Accuracy Graph’ to get below graph

Now click on ‘Classify Animal from Test Image’ button to get below output

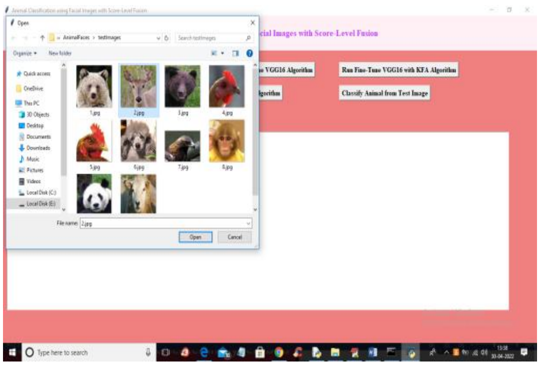

Fig.17.uploaded 2.jpg file

In above diagram I selected and uploaded 2.jpg file and then click on ‘Open’ button to get below output

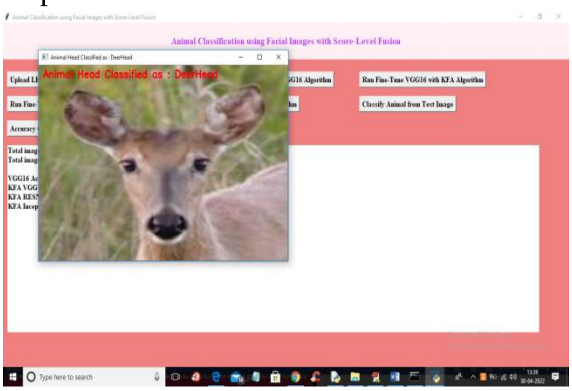

Fig.17.a.animal head classified Deer Head’

In above diagram on image red colour text saying animal head classified as Deer Head' similarly you can upload animal image and get classification output

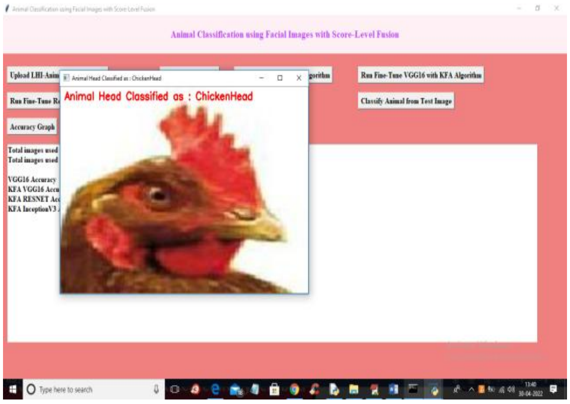

Fig.18.head classified Cock Head'

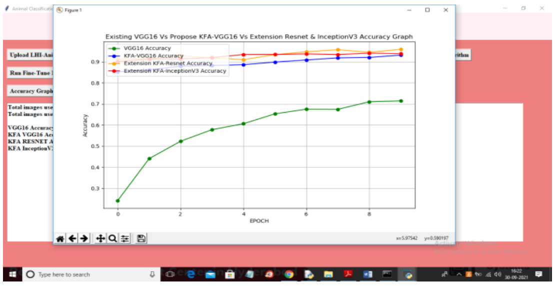

Fig.19. epoch vs accuracy

In above graph x-axis represents number of trains (epoch) and y-axis represents accuracy and in above graph green line represents VGG16 accuracy and blue line represents KFA-VGG16 accuracy and orange line represents KFA-Resnet accuracy and red line represents KFA-InceptionV3 accuracy and in above graph we can see extension algorithms RESNET and inceptionV3 has got more accuracy compare to VGG16.Note: here we split dataset into train and test randomly so accuracy may vary little for each execution

EXTENSION WORK

In propose paper author has used all deep learning algorithms such as VGG, RESNET with and without KFA features and in extension we have used advance XGBOOST algorithm with KFA which can train data accurately due to its inbuilt features such as optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. We are further optimizing KFA features by applying PCA algorithm and this optimization with XGBOOST giving accuracy up to 100%.

So in extension we have used PCA algorithm to optimize KFA features and then this optimized features are training with XGBOOST algorithm. Experiment with same dataset proving XGBOOST is better in performance compare to existing algorithms

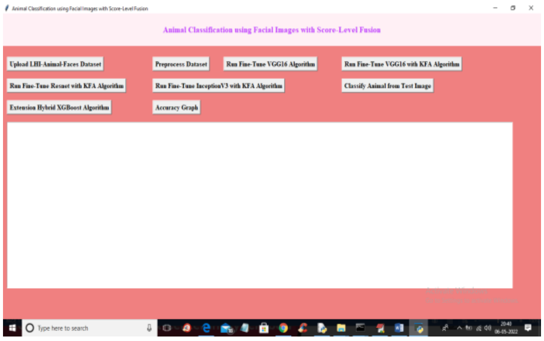

Fig.20.'Extension Hybrid XGBoost Algorithm'

In above diagram run all buttons as previous old project and then execute 'Extension Hybrid XGBoost Algorithm' button to train with XGBOOST and get below accuracy

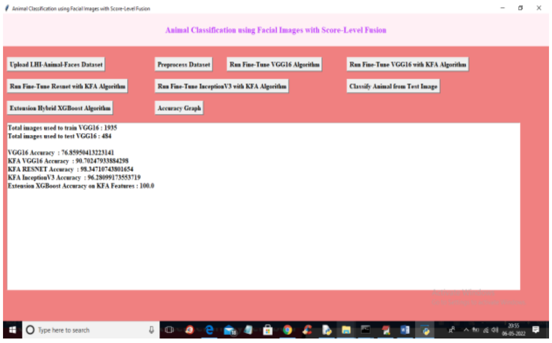

Fig.21. XGBOOST got 100% accuracy

In above diagram with extension XGBOOST we got 100% accuracy and this accuracy may vary little for every run as test data is split randomly

CONCLUSION

In this paper, we used all deep learning algorithms, like VGG and RESNET, with and without KFA features. We also used an advanced XGBOOST algorithm with KFA, which can train data accurately because it has features like an optimised distributed gradient boosting library that is designed to be highly efficient, flexible, and portable. We're making KFA features even better by using the PCA algorithm. This improvement, along with XGBOOST, brings the accuracy up to 100%.

BIBLIOGRAPHY

[1] Elson, J., Douceur, J., Howell, J., et al.: 'Asirra: a CAPTCHA that exploits interest-aligned manual image categorization'. Proc. ACM Conf. Computer and Communications Security (CCS), Alexandria VA, USA, 2007, pp. 366– 374

[2] Marcialis, G., Roli, F.: 'Score-level fusion of fingerprint and face matchers for personal verification under stress conditions'. 14th IEEE Int. Conf. Image Analysis and Processing ICIAP, DC, USA, 2007, pp. 259–264

[3] Elmir, Y., Elberrichi, Z., Adjoudj, R.: ‘Score-level fusion based multimodal biometric identification (fingerprint & voice)’. 6th Int. Conf. Sciences of Electronics, Technologies of Information and Telecommunications, Sousse, 2010, pp. 146–150
[4] Sim, H., Hishammuddin, A., Rohayanti, H., et al.: ‘Multimodal biometrics: weighted score-level fusion based on non-ideal iris and face images’, Expert Syst. Appl., 2014, 41, (11), pp. 5390–5404
[5] Patil, A., Bhalke, D.: ‘Fusion of fingerprint, palmprint and iris for person identification’. Int. Conf. Automatic Control and Dynamic Optimization Techniques (ICACDOT), Pune, 2016, pp. 960–963
[6] Takimoto, H., Mitsukura, Y., Fukumi, M., et al.: ‘Robust gender and age estimation under varying facial pose’, Electron. Commun. Jpn., 2008, 91, (7), pp. 32–40
[7] Schmid, C.: ‘Constructing models for content-based image retrieval’. Proc. 2001 IEEE Computer Society Conf. Computer Vision and Pattern Recognition, 2001. CVPR 2001, Kauai, USA, December 2001, pp. 11–39
[8] Ramanan, D., Forsyth, D.A., Barnard, K.: ‘Detecting, localizing and recovering kinematics of textured animals’. 2005 IEEE Computer Society Conf. Computer Vision and Pattern Recognition, San Diego, USA, June 2005, pp. 635–642
[9] Ramanan, D., Forsyth, D.A., Barnard, M.-K.: ‘Building models of animals from video’, IEEE Trans. Pattern Anal. Mach. Intell., 2006, 28, (8), pp. 1319– 1334
[10] Berg, T.L., Forsyth, D.A.: ‘Animals on the web’. 2006 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'06), NY, USA, 2006, pp. 1463–1470
[11] Penga, Z., Lia, Y., Caib, Z., et al.: ‘Deep boosting: joint feature selection and analysis dictionary learning in hierarchy’, Neurocomputing, 2016, 178, (20), pp. 36–45
[12] Afkham, H., Tavakoli, A., Eklundh, J., et al.: ‘Joint visual vocabulary for animal classification’. Int. Conf. Pattern Recognition, 2008. ICPR 2008, Tampa, FL, USA, 2008, pp. 1–4
[13] Si, Z., Zhu, S.-C.: ‘Learning hybrid image templates (HIT) by information projection’, IEEE Trans. Pattern Anal. Mach. Intell., 2012, 34, (7), pp. 1354– 1367
[14] Druzhkov, P.N., Kustikova, V.D.: ‘A survey of deep learning methods and software tools for image classification and object detection’, Pattern Recognit. Image Anal., 2016, 26, (1), pp. 9–15
[15] Krizhevsky, A., Sutskever, I., Hinton, G.: ‘Imagenet classification with deep convolutional neural networks’. Advances in Neural Information Processing Systems, Lake Tahoe, USA, 2012, pp. 1097–1105