1) Face classification and detection = Keras CNN Model

https://github.com/oarriaga/face_classification

Arriaga, O., Valdenegro-Toro, M. and Plöger, P., 2017. Real-time convolutional neural networks for emotion and gender classification. arXiv preprint arXiv:1710.07557.

2) Keras – Haar cascade - Based on Correa et al., (2016) - Emotion Recognition Using Deep Convolutional Neural Networks

https://github.com/atulapra/Emotion-detection

3) Emotion Detection from Tom and Jerry Videos – YOLO Algorithm, VGG-19 Convolutional Neural Networks

https://github.com/SurajSubramanian/EmotionDetection

4) Using LittleVGG for Emotion Detection

Load the pretrained model and run it on an image using the two lines of code below:

model = load_model("./emotion_detector_models/model.hdf5")
predicted_class = np.argmax(model.predict(face_image)

https://github.com/priya-dwivedi/face_and_emotion_detection/blob/master/src/EmotionDetector_v2.ipynb

GitHub Code – Face and Emotion Detection
https://github.com/priya-dwivedi/face_and_emotion_detection

5) Keras – Youtube Tutorial and Code from Github – Emotion Detection

https://github.com/neha01/Realtime-Emotion-Detection

https://youtu.be/DtBu1u5aBsc

6) Face Recognition and Emotion Detection

https://github.com/PraveenSaraswat/FREDS

---------- X ---------- X ----------

Animal Applications – Facial Recognition and Emotion Detection – Models

Marsot et al. (2019)

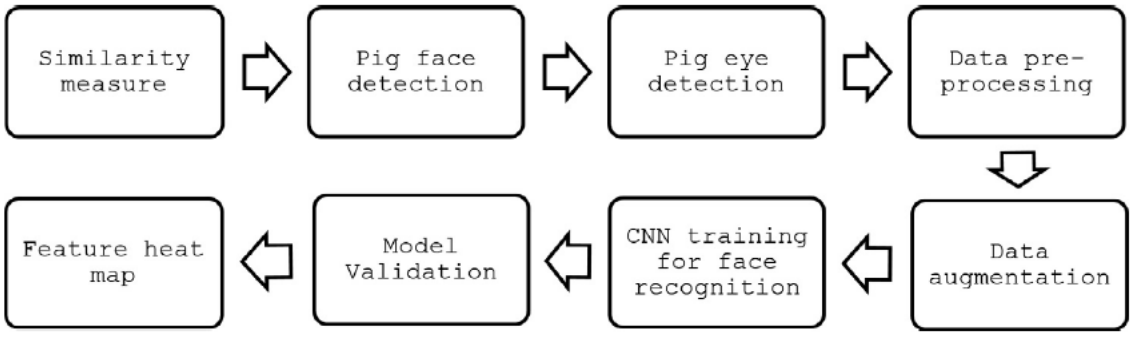

Fig. 1. The proposed framework of the adaptive pig face recognition solution.

Pig faces and eyes were detected by two Haar feature-based cascade classifiers and one shallow convolutional neural network (CNN) to extra high-quality images. Class activation maps generated by grad-CAM and saliency maps were utilized to visually understand how the discriminating parameters were learned by the neural network.

Wang et al., (2020) IOP Publishing

Structural similarity index (SSIM) is used to filter and select the picture data. Due to low insufficient data volume, 4 methods of image enhancement were used: rotation enhancement, blur enhancement, noise and flip enhancement. After this procedure, there were a total of 1087 images were used as training data for the neural network. A 19 layer deep network VGGNet network model with 5 group convolutions (2 layers of FC image features and 1 year of FC classification features) were used. After a set of processing steps, because the dataset is still less than 100,000 level data required to train the neural network, transfer learning method was employed.

Instead of using ImageNET trained models, neural network weights using the model parameters trained on the face dataset were used. Basic features of the bottom layer of the neural network were extracted. High-level uses the basic features to form local features and then the global features. Then, take the first two convolution groups of the frozen network, and fine-tune the neural network parameter of the next few layers.

Using Ubantu16.04+Tensorflow combined with GPU acceleration calculation, the visualization of the result was realized using the matplotlib library. Cosine similarity between the feature vector of the image to be identified and the registered feature vector in the face database was considered for the same animal. Comparison between the ImageNet dataset pre-trained network + fine-tuned network, VGGFace Pre-trained, VGGFace Pre-trained + fine-tuned network provided the classification results with over 91%, 95% and 93% accuracies.

Mahmoud et al., (2018)
Viola-Jones object detection framework was employed for the frontal face detection of sheep. Booting procedure was adopted to achieve larger number of training samples. Dlib implementation of a DNN-based MMOD algorithm was also used. MMOD was found to be highly effective as it optimizes the sub-windows of each image and boosts the performance of the face detector. For facial landmark localization, cascaded pose regression (CPR) scheme was used along with the TIF (Yang et al. 2015) model. TIF model was found to be better as TIF can draw features from a larger area. Feature-wise normalisation was applied on the sheep faces and normalised separately to extract the ears, eyes and noses from sheep face images. Dlib (King, 2009) was implemented in the histogram of oriented gradients (HOG). Each block of HOG stands of a 31 dimensional vector. 4 normalisation masks are applied on top of the 9-orientational histogram, followed by PCA reduction. HOG descriptors are visualized showing the block dimensions of the ear, eye and nose. Support vector machines (SVMs) were then used to train the separate classifiers for each facial feature.

Hansen et al., (2018)
Fisherfaces, VGG-Face pre-trained face convolutional neural network (CNN) model using Class Activated Mapping using Grad-CAM to discriminate between face pigs. Data cleaning was done by structural similarity index measure to determine and avoid the similar images between train and test data partitions.

Corkery et al., (2007)
Analysis of sheep face images by independent components technique. Using cosine distance classifier.