Facial Emotion Recognition Recap

There are many research articles dealing with the topic of FER and it is still a challenge to get high precision results (over 95%). The main tries are using either deep learning algorithm or machine learning algorithm such as SVM or KNN. This article summarizes all the trends until 2022 and it gives the red thread to follow.

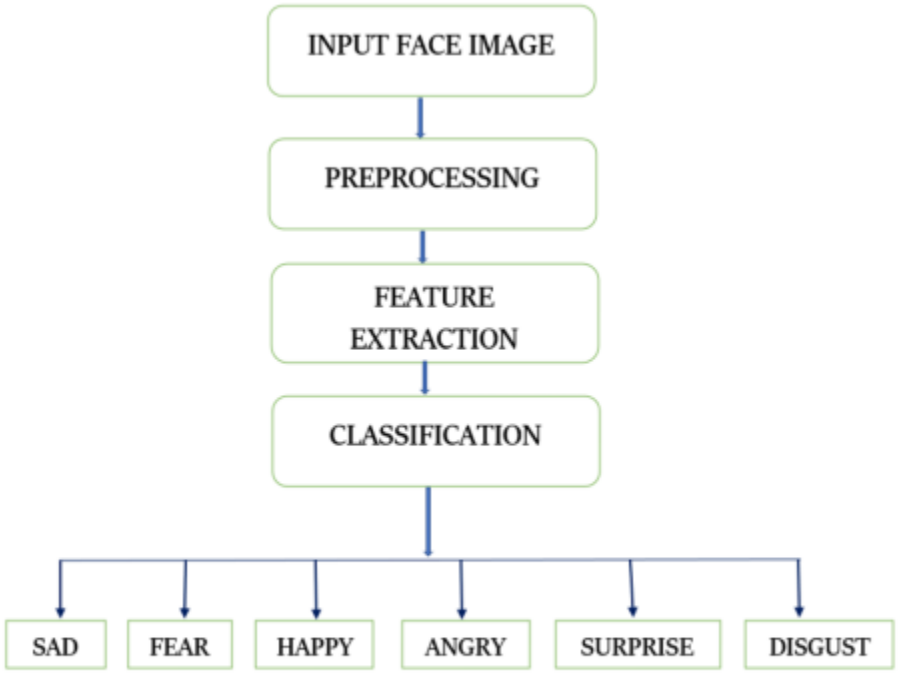

Fig. 2 Block diagram representation of FER system (Michael Revina and Sam Emmanuel 2021)

1 - Dataset

There are many datasets available online. The most used and interesting one is FER2013, composed of 30K+ face images in grayscale 48x48 pixels. The pictures are sorted into different folders according to the emotion represented: happiness, fear, sadness, surprise, anger, disgust and neutral. Other datasets can be taken from CK+ or RAFD-DB. We could also create our own dataset or extend the one we have with small translations, rotations, filters, or changing

2 – Pre-processing

Now we have a dataset, we need to make sure that the model will only be using images with the same format. So, if we are using grayscale, 48x48 pixels, face centered images then we need to create functions that does that. Resizing and turning an image to black and white is not challenging. The most difficult part is to crop the face. This can be done using existing algorithms.

3 – Facial feature extraction

Now that we have a uniform dataset, we need to find what our model will be fed with. This is the first important part of the project, and several methods exist. One of them is to crop the various parts of the face such as the eyes, the nose or the mouth and label each part with the corresponding emotion. That method is called Zoning. These extractions can be done with different python libraries like dlib and cv using Haar Cascade algorithm. Another technique is to extract a face mesh. This can be profitable to just give the mesh to the model because it would reduce the computing time a lot (for 70 nodes mesh). The mesh can be extracted thanks to the MediaPipe library, or we can create our own by using image processing algorithms such as SIFT or SURF allowing to get the regions of interest (ROI) or at least facial landmarks.

4 – Model and training

At this point, we know what our model is going to use but do not know what it is yet. In fact, there are two schools. The first one just applies a deep learning model. Then we can either use a pre-trained neural network like VGG, ResNet or Inception or create our own one. The other school only applies different machine learning algorithms (SVM, KNN, Decision tree, Random Forest, K-means *(suggestion?)*) as shown in this article. They give equivalent results according to the methods, model and dataset used.

5 – Evaluation and Performance

Once we have the results, we need to evaluate our model. The first way is to have a look at the confusion matrix that shows the rate of good prediction for each emotion. Then we can calculate the following indicators:

$$Accuracy = \frac{No.\ of\ \ correctpredictions}{Total\ no/,\ .of\ predictions} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{Misclassification} = \frac{No.\ of\ \ incorrect\ predictions}{Total\ no.\ of\ predictions} = \frac{FP + FN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$SensitivityorRecall = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

$$F1\_Score = \frac{2 \times precision \times recall}{precision + recall}$$

We can use these indicators to compare our results as shown in [this article].

The topic of emotion recognition has been and is still studied by a lot of researchers. Thus, it is quite difficult to find something that is not done yet but, I did not find any article comparing the results depending on features used (pure image or zoning or mesh or landmarks) and moreover there is no real comparison between machine learning's result and deep learning's (neural network).