IJMTST is UGC Approved Journal with Journal Id: 43137

International Journal for Modern Trends in Science and Technology, 8(S08): 80-84, 2022

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST08S0815
Available online at: https://www.ijmtst.com/vol8si08.html

Suppressing uncertainties with Large-Scale Facial Expression Recognition

Indukuri Rahul Varma | Satya Shekar Varma P

Computer Science and Engineering, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India

To Cite this Article
Indukuri Rahul Varma and Satya Shekar Varma P. Suppressing uncertainties with Large-Scale Facial Expression Recognition. International Journal for Modern Trends in Science and Technology 2022, 8(S08), pp. 80-84. https://doi.org/10.46501/IJMTST08S0815

Article Info
Received: 26 May 2022; Accepted: 24 June 2022; Published: 28 June 2022.

ABSTRACT

Annotating a qualitative large-scale facial expression dataset is extremely difficult due to the uncertainties caused by ambiguous facial expressions, low-quality facial images, and the subjectiveness of annotators. These uncertainties lead to a key challenge of large-scale Facial Expression Recognition (FER) in deep learning era. To address this problem, a simple yet efficient Self-Cure Network (SCN) is proposed which suppresses the uncertainties efficiently and prevents deep networks from over-fitting uncertain facial images. Specifically, SCN suppresses the uncertainty from two different aspects: One aspect is a self-attention mechanism over mini batch to weight each training sample with a ranking regularization and other aspect is a careful relabeling mechanism to modify the labels of these samples in the lowest-ranked group. Experiments on synthetic FER datasets and collected Web-emotion dataset validate the effectiveness of proposed method. Results on public benchmarks demonstrate that our SCN outperforms current state-of-the-art methods with 88.14% on RAF-DB, 60.23% on AffectNet, and 89.35% on FERPlus. However, for the large-scale FER datasets collected from the Internet, it is extremely difficult to annotate with high quality due to the uncertainties caused by the subjectiveness of annotators as well as ambiguous in-the-wild facial images.

KEYWORDS: *Facial-Expression Recognition, Self-Cure Network, MTCNN Algorithm, Web-emotion, Face Detection*

1. INTRODUCTION

Facial expression is one of the most natural, powerful and universal signals for human beings to convey their emotional states and intentions. Automatically recognizing facial expression is also important to help the computer understand human behaviour and to interact with them. In the past decades, researchers have made significant progress on facial expression recognition (FER) with algorithms and large-scale datasets, where datasets can be collected in laboratory or in the wild, such as CK+, MMI, OuluCASIA, SFEW/AFEW, FERPlus, AffectNet, EmotioNet, RAF-DB, etc. However, for the large-scale FER datasets collected from the Internet, it is extremely difficult to annotate with high quality due to the uncertainties caused by the subjectiveness of annotators as well as ambiguous in-the-wild facial images. The uncertainties increase from high-quality and evident facial expressions to low quality and micro expressions.

These uncertainties usually lead to inconsistent labels and incorrect labels, which are suspending the progress of large-scale Facial Expression Recognition (FER), especially for the one of data-driven deep learning-based FER. Generally, training with uncertainties of FER may lead to the following problems. First, it may result in over-fitting on the uncertain samples which may be mislabeled. Second, it is harmful for a model to learn useful facial expression features. Third, a high ratio of

incorrect labels even makes the model di-convergence in the early stage of optimization. To address these issues, a simple yet efficient method is proposed, termed as Self-Cure Network (SCN), to suppress the uncertainties for large-scale facial expression recognition. The SCN consists of three crucial modules: self-attention importance weighting, ranking regularization, and noise re-labelling.

2. Literature Survey

"Entropy driven feature selection for facial expression recognition based on 3-D facial feature distances" is developed by Kamil Yurtkan; HamitSoyel; Hasan Demirel. It was published in 2015. They have used SVM classifier. It has drawbacks like Choosing the right kernel function is not easy[1].

"A Review of Face Recognition Technology" is developed by Lixiang Li Xiaohui Mu, Siying Li AndHaipeng Peng et.al. It was published in 2020. They have used Biometric Algorithm Ada Boost. It has drawbacks like it is not suitable for real time facial recognition [2].

"Face Detection and Recognition Using OpenCV" is developed by Maliha Khan; Sudarshan Chakraborty; Rani Satya; ShavitKhepra et.al. It was published in 2019. They have used Principal Component Analysis, Local Binary Patterns. It has drawbacks like without resolution it does not work [3].

"Facial Expression Recognition expression from Image Sequence Based on LBP and Taylor Expansion" is developed by Qin Zhao, Yuanyuan Ding et.al. It was published in 2017. They have used Local Binary Patterns. It has drawbacks like Works poor in uncontrolled environment [4].

3. Design and Methodology

Facial expression recognition is a well-known activity in the domain of human–computer interaction and computer vision. In this project, face detection algorithm is applied on the images to get the facial part only; then, local binary pattern (LBP) operator is used to get the facial features.

At present, traditional facial expression recognition methods of convolutional neural networks are based on local ideas for feature expression, which results in the model's low efficiency in capturing the dependence between long-range pixels, leading to poor performance for facial expression recognition.

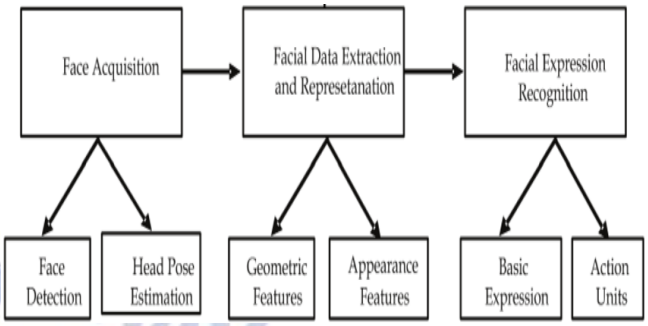

Figure 2.1: System Architecture of proposed method

4. Dataset

This project introduces recent advances in facial expression analysis and recognition. The first part discusses general structure of AFEA systems. The second part describes the problem space for facial expression analysis. This space includes multiple dimensions: level of description, individual differences in subjects, transitions among expressions, intensity of facial expression, deliberate versus spontaneous expression, head orientation and scene complexity, image acquisition and resolution, reliability of ground truth, databases, and the relation to other facial behaviours or non-facial behaviours. It is noted that most work to date has been confined to a relatively restricted region of this space. The last part of this project is devoted to a description of more specific approaches and the techniques used in recent advances. They include the techniques for face acquisition, facial data extraction and representation, facial expression recognition, and multimodal expression analysis. The project concludes with a discussion assessing the current status, future possibilities, and open questions about automatic facial expression analysis.

A. Data pre-processing

Data pre-processing is a data mining technique that involvestransforming raw data into an understandable format. Real-world data is often incomplete, inconsistent, and/or lacking incertain behaviour or trends, and is likely to contain manyerrors. Data pre-processing is a proven method of resolvingsuchissues.

STEP 1:IMPORTINGTHELIBRARIES

The following are screenshots of the coding pat of proposed system. Figure 4.1.1, figure 4.1.2 and figure 4.1.3 shows the coding part for importing libraries.

```
import random
new_file = open("0.2noise_train.txt","w+")
with open("train.txt","r") as file:
        for line in file:
                line = line.strip()
                img_path, label = line.split(' ', 1)
                number = random.uniform(0,1)
                new_label = random.randint(0,6)
                if number <= 0.2:
                        while(1):
                                new_label = random.randint(0,6)
                                if new_label != int(label):
                                        new_file.write(img_path + ' ' + str(new_label) +'\n')
                                        break
                else:
                        new_file.write(img_path + ' ' + str(label) +'\n')
```

Figure 4.1.1: Screenshot of Imported in Proposed System

```
from fer import Video
from fer import FER
import os
import sys
import pandas as pd

# Put in the location of the video file that has to be processed
location_videofile = "/content/Video_One.mp4"

# Build the Face detection detector
face_detector = FER(mtcnn=True)
# Input the video for processing
input_video = Video(location_videofile)

# The Analyze() function will run analysis on every frame of the input video.
# It will create a rectangular box around every image and show the emotion values next to that.
# Finally, the method will publish a new video that will have a box around the face of the human with live emotion values.
processing_data = input_video.analyze(face_detector, display=False)

# We will now convert the analysed information into a dataframe.
# This will help us import the data as a .CSV file to perform analysis over it later
vid_df = input_video.to_pandas(processing_data)
vid_df = input_video.get_first_face(vid_df)
vid_df = input_video.get_emotions(vid_df)

# Plotting the emotions against time in the video
pltfig = vid_df.plot(figsize=(20, 8), fontsize=16).get_figure()

# We will now work on the dataframe to extract which emotion was prominent in the video
angry = sum(vid_df.angry)
disgust = sum(vid_df.disgust)
fear = sum(vid_df.fear)
```

Figure 4.1.2: Screenshot of Imported in Proposed System

```
happy = sum(vid_df.happy)
sad = sum(vid_df.sad)
surprise = sum(vid_df.surprise)
neutral = sum(vid_df.neutral)

emotions = ['Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Neutral']
emotions_values = [angry, disgust, fear, happy, sad, surprise, neutral]

score_comparisons = pd.DataFrame(emotions, columns = ['Human Emotions'])
score_comparisons['Emotion Value from the Video'] = emotions_values
score_comparisons
```

Figure 4.1.3: Screenshot of Imported in Proposed System

STEP 2: INSTALLING PIP

```
pip install fer

Collecting fer
  Downloading fer-21.0.5-py3-none-any.whl (810 kB)
     |████████████████████████████████| 810 kB 3.7 MB/s
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages (from fer) (4.62.3)
Requirement already satisfied: keras>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from fer) (2.7.0)
Requirement already satisfied: opencv-contrib-python in /usr/local/lib/python3.7/dist-packages (from fer) (4.1.2.30)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from fer) (1.1.5)
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from fer) (2.23.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages (from fer) (3.2.2)
Collecting mtcnn>=0.1.1
  Downloading mtcnn-0.1.1-py3-none-any.whl (2.3 MB)
     |████████████████████████████████| 2.3 MB 56.2 MB/s
Requirement already satisfied: opencv-python>=4.1.0 in /usr/local/lib/python3.7/dist-packages (from mtcnn>=0.1.1->fer) (4.1.2.30)
Requirement already satisfied: numpy>=1.14.5 in /usr/local/lib/python3.7/dist-packages (from opencv-python>=4.1.0->mtcnn>=0.1.1->fer) (1.19.5)
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib->fer) (2.8.2)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib->fer) (0.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib->fer) (1.3.2)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib->fer) (3.0.6)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.1->matplotlib->fer) (1.15.0)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages (from pandas->fer) (2018.9)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from requests->fer) (2021.10.8)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (from requests->fer) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python3.7/dist-packages (from requests->fer) (1.24.3)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests->fer) (2.10)
Installing collected packages: mtcnn, fer
Successfully installed fer-21.0.5 mtcnn-0.1.1
```

Figure 4.2. Installing of pip

Figure 4.2 Installing of pip. The pip install <package> command always looks for the latest version of the package and installs it. It also searches for dependencies listed in the package metadata and installs those dependencies to ensure that the package has all the requirements it needs.

STEP 3: LOAD THE DATASET

Load the dataset which is the image of a emotion of a person and then save it under the folder of sample_data. Then process the image with help of MTCNN algorithm. For example, we will be taking the image of happy person as shown in figure 4.3.

Figure 4.3: Happy Image

STEP 4: APPLY MTCNN ALGORITHM

Multi-Tasking Convolutional Neural Network (MTCNN) is a subfield machine learning in which multiple learning tasks are solved at the same time, while exploiting commonalities and differences across

tasks. This can result in improved learning efficiency and prediction accuracy for the task-specific models, when compared to training the models separately. It does this by learning tasks in parallel while using a shared representative; what is learned for each task can help other tasks be learned better.

In the classification context, MTL aims to improve the performance of multiple classification tasks by learning them jointly. One example is a spam-filter, which can be treated as distinct but related classification tasks across different users. To make this more concrete, consider that different people have different distributions of features which distinguish spam emails from legitimate ones, for example an English speaker may find that all emails in Russian are spam, not so for Russian speakers. Yet there is a definite commonality in this classification task across users, for example one common feature might be text related to money transfer.

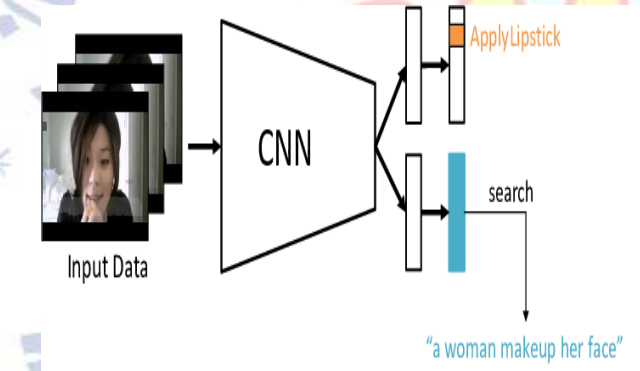

Figure 4.4: Multi-Tasking Convolutional Neural Network

```
from fer import FER
import cv2

img = cv2.imread("sample_data/happy.jpg")
detector = FER()
detector.detect_emotions(img)
```

```
[{'box': (293, 216, 375, 375),
  'emotions': {'angry': 0.0,
   'disgust': 0.0,
   'fear': 0.01,
   'happy': 0.87,
   'neutral': 0.08,
   'sad': 0.03,
   'surprise': 0.0}}]
```

Figure 4.5: Output for happy image input

5. CONCLUSION

As is quite evident after plenty of research and deliberation, gaining insight on what a person may be feeling is very valuable for many reasons. The future scope of this field is visualized to be practically limitless, with more futuristic applications visible on the horizon especially in the field of Security. Proposed facial expression recognition system, utilizing neuro-fuzzy architecture is 71.7% accurate, which is approximately the level of accuracy expected from a support vector machine approach as well. Every system has its limitations. Although this implementation of facial expression recognition may perform less than entirely accurate as per the end users' expectations, it is envisioned to contribute significantly to the field, upon which similar work can be furthered and enhanced. Main aim in forming such a system is to form a standard protocol that may be used as a component in many of the applications that will no doubt require an emotion based HCI. Empowering computers in this way has the potential of changing the very way a machine —thinks. It gives them the ability to understand humans as _feelers 'rather than _thinkers '. This in mind, this system can even be implemented in the context of Artificial Intelligence. As part of the relentless efforts of many to create intelligent machines, facial expressions and emotions have and always will play a vital role.

6. FUTURE SCOPE

As human facial expression recognition is a very elementary process, it is useful to evaluate the mood or emotional state of a subject under observation. As such, tremendous potential lies untapped in this domain. The basic idea of a machine being able to comprehend the human emotive state can be used in innumerable scenarios, a few are:

- The ability to detect and track a user's state of mind has the potential to allow a computing system to offer relevant information when a user needs help – not just when the user requests help, for instance, the change in the Room Ambience by judging the mood of the person entering it.
- Help people in emotion-related research to improve the processing of emotion data. ¬

Applications in surveillance and security. For instance, computer models obtained up to 71% correct classification of innocent or guilty participants based on the macro features extracted from the video camera footage.

- In this regard, lie detection amongst criminal suspects during interrogation is also a useful aspect in which this system can form a base. It is proven that facial cues often can give away a lie to the trained eye.
- Patient Monitoring in hospitals to judge the effectiveness of prescribed drugs is one application to the Health Sector. In addition to this, diagnosis of diseases that alter facial features and psychoanalysis of patient mental state are further possibilities.
- Clever Marketing is feasible using emotional knowledge of a patron and can be done to suit what a patron might need based on his/her state of mind at any instant.
- Detecting symptoms such as drowsiness, fatigue, or even inebriation can be done using this system. Thus, by helping in the process of Driver Monitoring, this system can play an integral role in reducing road mishaps to a great extent.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] Piotr Dalka, Andrzej Czyzewski, ―Human-Computer Interface Based on Visual Lip Movement and Gesture Recognition‖, International Journal of Computer Science and Applications, ©Technomathematics Research Foundation Vol. 7 No. 3, pp. 124 - 139, 2010 International Journal of Computer Science and Applications, ©Technomathematics Research Foundation Vol. 7 No. 3, pp. 124 - 139, 2010.

[2] M. Pantic, L.J.M. Rothkrantz, ―Facial Gesture Recognition in face image sequences: A study on facial gestures typical for speech articulation‖ Delft University of Technology.

[3] Claude C. Chibelushi, Fabrice Bourel, ―Facial Expression Recognition: A Brief Overview‖, 1-5 ©2002.

[4] AysegulGunduz, Hamid Krim ―Facial Feature Extraction Using Topological Methods‖ © IEEE, 2003.

[5] Kyungnam Kim, ―Face Recognition Using Principal Component Analysis‖

[6] Abdallah S. Abdallah, A. Lynn Abbott, and Mohamad Abou El-Nasr, ―A New Face Detection Technique using 2D DCT and Self Organizing Feature Map‖, World Academy of Science, Engineering and Technology 27 2007.

[7] Sander Borsboom, SanneKorzec, Nicholas Piel, ―Improvements on the Human Computer Interaction Software: Emotion‖, Methodology, 2007, 1-8.

[8] A. Mehrabian, Communication without Words, Psychology Today 2 (4) (1968) 53-56.

[9] M. Pantic, L.J.M. Rothkrantz, ―Expert System for Automatic for Automatic Analysis of Facial Expressions‖ Image and Vision Computing 18 (2000) 881-905.

[10] Li Chaoyang, Liu Fang, XieYinxiang, ―Face Recognition using Self-Organizing Feature Maps and Support Vector Machines‖, Proceedings of the Fifth ICCIMA, 2003.