**Entertainment scoring system based on facial expression detection**

1. **Project Overview**

In the entertainment industry, judging the quality of an artist’s performance is an extremely subjective task, wherein, the actual performance of the artist can often be misjudged because of the artist’s reputation and many atmospheric factors.

Keeping this in mind, a scoring system where subjective factors are taken and statistics pertaining to the performance in regards of, the number of people that portray a certain emotion, for example, happiness, sadness and laughter can truly help those who host artists and also the artists themselves in getting solid subjective feedback.

Our solution involves the use of Computer Vision technologies to enable this. Using Live Video of the audience, we try to best compute the number of people showing emotions useful for our statistics.

In the end statistic, we can see the average score over the duration of the performance and therefore give the artist useful feedback.

Also, by observing certain time frames where the emotional statistic stands out, we can give the artist useful feedback on the quality of their material.

1. **Purpose**

* To provide artists and hosts valuable subjective feedback depending on the audience reaction over the duration of a performance.
* To recognize the quality of an artist’s performance over their reputation.
* To analyze the reactions or facial expressions of the audience during showcase of movies and use this information to improve new movies on facts that the audience feels amusing and what doesn’t.

1. **Related Work**

# **Target Audience Response Analysis in Out-of-home Advertising Using Computer Vision**

A. Costache, D. Popescu, S. Mocanu and L. Ichim, "Target Audience Response Analysis in Out-of-home Advertising Using Computer Vision," *2020 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, Bucharest, Romania, 2020, pp. 1-6, doi: 10.1109/ECAI50035.2020.9223134.

This paper aims to analyze the response of the target audience to out-of-home advertising panels mounted in display windows. They track people passing in front of the panels and, if they approach the panel, looking directly at it, they analyze their facial movements, to see whether the advert appeals to them or not. The approach is based on video data gathering using a static IP camera, storage and analysis, aiming to deliver real-time results. They use neural networks and support vector machines to determine facial micro expression. Results are correlated with information about the adverts displayed to be of use to advertising providers.

* 1. **Factorized Variational Autoencoders for Modeling Audience Reactions to Movies**

*Z. Deng et al., "Factorized Variational Autoencoders for Modeling Audience Reactions to Movies," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 6014-6023, doi: 10.1109/CVPR.2017.637.*

This paper uses facial imagery from a moviegoing audience to detect emotions and predict their reactions for the rest of the movie.

The system was able to predict emotions after only about 10 minutes of “warmup” (data gathering) and that too, reliably. The algorithm was able to predict smiles and laughs from the movie watching audience.

* 1. **Real-time surveillance and evaluation system for audience reaction to meeting sessions**

[*Jingshan Tang*](https://www.spiedigitallibrary.org/profile/notfound?author=Jingshan_Tang) *and* [*Jun Ohya*](https://www.spiedigitallibrary.org/profile/notfound?author=Jun_Ohya) *"Real-time surveillance and evaluation system for audience reaction to meeting sessions", Proc. SPIE 3837, Intelligent Robots and Computer Vision XVIII: Algorithms, Techniques, and Active Vision, (26 August 1999);* [*https://doi.org/10.1117/12.360324*](https://doi.org/10.1117/12.360324)

This paper describes a visual surveillance system for evaluating the audience's reaction in meeting sessions. The system, which works in real-time, can recognize and evaluate the reaction of the audience. It is mainly composed of three subsystems. The first subsystem is a face detection and head motion segmentation system which is used to detect the face from complex background and segment the head motion into different units, with each unit including different information about the audience's reaction. The second subsystem is a gesture and pose recognition system which can recognize the gestures and the poses of human heads. The third subsystem is an evaluation system which is used to evaluate the reaction of the audience by using the recognition result in the second system.

1. **Method and model:**

You can easily identify faces of people or groups in photos or videos yourself but making any code or algorithm do that for you will be extremely useful. Using this information, we can create powerful applications. But first we need information about the person’s face, like position, whether the mouth is opened or closed, whether the eyes are opened, closed, looking up and etc. This technology of face detection is used in applications to detect faces from digital images and videos. Also, just detecting the face will not provide enough information so we need more information about the face, i.e., whether a person smiles or laughs. In short, facial expressions too give us information. In this project, we find facial landmarks and thus determine the facial expression of people out of it. We used one of powerful python computer vision library Dlib, capable of giving you 68 points (landmarks) of the face.

Here is list of facial expression found in this project:

1. Happy

2. Sad

3. Normal

4. Surprised

5. Terrified

Algorithmic steps:

1) The RGB image is read from the local storage. The image is then converted to greyscale so that dlib shape predictor function can run on it.

2) The image is then converted to greyscale so that dlib shape predictor function can run on it. Firstly, we will get the predictor path of “shape\_predictor\_68\_face\_landmarks.dat”, a pre-trained model file of iBUG 300-W dataset. Secondly, we need to make a predictable object of model so we call the facial landmark predictor function dlib.shape\_predictor() with a predictor path as parameter so that it can detect 68 facial landmarks on test images.

To detect all faces in image or frame, we call a frontal face detector function dlib.get\_frontal\_face\_detector(). This is a pre-trained face detector based on Histogram of Oriented Gradients (HOG) features.

To get the coordinates of all faces present in the image, we call detector.run(). It returns a list of rectangle objects which outline every face in the image.

3) 3 copies of the original image are created. Here is a list depicting the use of each one.

a) First image to outline every detected face in the image.

b) Second image to mark 68 facial landmarks on each detected face.

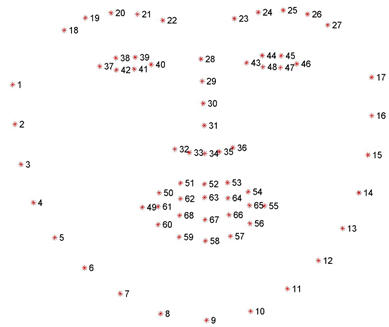
c) Third image to show the predicted count of facial expressions for all detected faces.

4) A ‘expressions’ list with 5 zeros is created. This list will store the numbers of each facial expression found in the image. These 5 expressions are happy, sad, normal, surprised and terrified.

5) All the tasks described in the last step are done using a loop that is run for a list of rectangle objects returned by detector.run().

1. Firstly, we call rect\_to\_box() function and pass each rectangle object one by one and so it creates green rectangles on image depicting the detected faces in image with a number which describes the order in which the faces were detected.
2. Secondly, we call annotate\_landmarks() which draws 68 facial landmarks of each detected face on the image. Using predictor() function, we create a list of coordinates of each facial landmark to be able to draw 68 facial landmarks on image and also required predicted facial expressions.
3. Thirdly, we call predict\_facial\_expression() function to predict the facial expression of each detected face in an image. The number of each 5 facial expressions is marked on top left of this image. The mathematical aspect of prediction is explained in the next step.

6) The prediction of facial expression is done using 4 facial landmarks of mouth which are 61, 63, 65, 67 (index 60, 62, 64, 66 of ‘landmarks’ python list).



**Figure 1**: Visualizing the 68 facial landmark coordinates

a) Below are landmark points numbers and its description.

i) 61: Left inner intersection of upper and lower lip

ii) 63: Middle inner part of upper lip

iii) 65: Right inner intersection of upper and lower lip

iv) 67: Middle inner part of lower lip

b) Using an imaginary line between landmarks 61 and 65, find the distance from perpendicular distance between upper and lower lips (landmarks 63 and 67 respectively), and are stored in dist1 and dist2 variables. We also find distances between landmarks 61 and 65, and stored in the left\_to\_right variable. Values of these 3 variables are then manipulated and compared using a nested if-else statement.

c) Below is the list of comparisons done using nested if-else statements.

i) dist1 > 0 and dist2 > 0: The person is making a sad/ frowning face because landmarks of both lips are on top of the imaginary line and are making shape like a downward parabola.

ii) dist1 > 0 and dist2 > 0: The person is making a smiling face because landmarks of both lips are on the bottom of the imaginary line and are making a shape like an upward parabola.

iii) dist1 > abs(dist2) and (dist1 - dist2) / left\_to\_right > 0.1: The person's face is terrified/ horrified because landmarks of both lips are on top of the imaginary line and are making shape like a downward parabola. But unlike (i), the distance between the lips is appreciable.

iv) dist1 > abs(dist2) and (dist1 - dist2) / left\_to\_right < 0.1: The person's face is deemed normal because the distance between the lips is negligible.

v) dist1 < abs(dist2) and (dist1 - dist2) / left\_to\_right < 0.1: The person's face is deemed normal again because the distance between the lips is negligible.

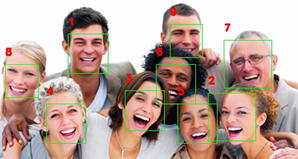
vi) dist1 < abs(dist2) and (dist1 - dist2) / left\_to\_right >= 0.1 and dist1 + abs(dist2)) / left\_to\_right < 0.6: The person is making a smiling/laughing face because there is an appreciable distance between lips.

vii) dist1 < abs(dist2) and (dist1 - dist2))) / left\_to\_right > 0.6: The person is making a surprised/ astonished face because there is an appreciable distance between lips. But unlike (vi), the lower jaw of mouth is pulled making left\_to\_right distance less and distance between lips more thus helping in more positive ((dist1 - dist2) / left\_to\_right) value.

7) All three images are then displayed to the user.

**5. Results:**

Below are some images showing detected faces and amount and type of facial expressions:

**Figure 2**: Displaying the detected faces with number of 5 different facial expressions

**Figure 3**: Displaying the detected faces with number of 5 different facial expressions

**Figure 4**: Displaying the detected faces with number of 5 different facial expressions

**Figure 5**: Displaying the detected faces with number of 5 different facial expressions

**6. Conclusion and future scope:**

By performing experiments on multitudes of images which vary in factors such as number of people, orientation of faces, skin tones and crowd density, we see that the algorithm has a satisfying degree of accuracy which could indeed be used in the planned applications.

The algorithm can be improved by using other facial landmarks of mouth and also taking account of eyes and eyebrows landmarks as they also play a role in actual estimation of facial expressions.

Our research was limited to the computer vision aspect of this project but using the above concept. This technique can be combined by voice feedback by the audience to predict some entertainment content most accurately as possible. In CCTV, facial expression predictions can be used to detect any suspicious activities on streets, shopping places, restricted areas and especially in prisons.

**References :**

[1] A. Costache, D. Popescu, S. Mocanu and L. Ichim, "Target Audience Response Analysis in Out-of-home Advertising Using Computer Vision," *2020 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, Bucharest, Romania, 2020, pp. 1-6, doi: 10.1109/ECAI50035.2020.9223134.

[2] *Z. Deng et al., "Factorized Variational Autoencoders for Modeling Audience Reactions to Movies," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 6014-6023, doi: 10.1109/CVPR.2017.637.*

[3] [*Jingshan Tang*](https://www.spiedigitallibrary.org/profile/notfound?author=Jingshan_Tang) *and* [*Jun Ohya*](https://www.spiedigitallibrary.org/profile/notfound?author=Jun_Ohya) *"Real-time surveillance and evaluation system for audience reaction to meeting sessions", Proc. SPIE 3837, Intelligent Robots and Computer Vision XVIII: Algorithms, Techniques, and Active Vision, (26 August 1999);* [*https://doi.org/10.1117/12.360324*](https://doi.org/10.1117/12.360324)