# *Gender Encoder By Human Smile*

***By focusing on Testing strategies by using different classifiers.***

***Muhammad Waqar Younus***

***Sunbul***

***Zeeshan Qaisar***

# ABSTRACT

# Gender classification has become a topic of great interest to the visual computing research community in recent times. This is due to the fact that computer-based gender recognition has multiple applications including, but not limited to, face perception, age, ethnicity, identity analysis, video surveillance and smart human computer interaction. In our project, we implement a machine learning approach for efficient identification of gender purely from the dynamics of a person’s smile. Thus, we show that the complex dynamics of a smile on someone’s face bear much relation to the person’s gender. To do this, we first formulate a computational framework that captures the dynamic characteristics of a smile. The face geometric model relies from a set of facial landmarks for extracting the relevant feaures.We have used predictor that takes as input a face image and output the position of 68 facial landmarks expressed as pairs of Certesian coordinates(x,y). For machine classification, we have utilized Random Forest, Naive Bayes and Support Vector Machine (SVM) algorithm. To verify the accuracy of our approach, we have tested our algorithms on dataset , namely the CK+, consisting of a total of 60 subjects. In which 30 are males and 30 are females . As a result, using the *Random Forest and SVM* algorithm, we achieve 83% accuracy with test size 0f 20% and 63% accuracy with test size of 30% respectively.

# We have also applied some major techniques of Software Dependability like Clustering, Mining and Testing and used some Machine Learning Algorithms.

# 1 – Introduction

It is often said that the face is a window to the soul. Bearing a image of this nature in mind, one might find it interesting to understand, if any, how the physical, behavioral as well as emotional characteristics of a person could be decoded from the face itself. With the increasing power of machine learning techniques, it`s become truly to address such questions through the development of appropriate computational frameworks.

Computational frameworks for human face analysis have recently found their way into many application areas. These include computer vision, psychology, biometrics, health care etc. The appealing, and the practical, nature of such face analysis techniques, are highlighted by the wealth of information it can provide in a non-invasive manner. Computer-based analysis of the human face can provide strong and useful cues for personal attributes such as age, ethnicity and more appropriately gender. Gender classification, in this sense, can, for example, aid as an advantageous biometric feature in order to improve the accuracy of determining an identity, especially in the presence of limited information on a subject. Recent research into gender classification has faced challenging hurdles, mainly due to the reliance of static data in the form of facial images. There are many inherent issues when looking for gender clues in appearance-based facial analysis. These include variability of lighting conditions, pose and occlusions. In this regard, in this project, we departed from such appearance-based analysis of facial images. Instead, we consider the analysis of the dynamic face, in particular, the dynamics of the smile, for clues of gender.

Hence, our project is concerned with the identification of gender from the dynamic behaviour of the face. Equally importantly, we seek to answer the crucial question of whether gender is encoded in the dynamics of a person’s smile. We specifically focus on studying the smile as it is considered to be a rich, complex and sophisticated facial expression, formed through the synergistic action of emotions. Various studies show that there are differences in smiles between males and females, i.e. females tend to bear more expressive smiles than males.

Based on the findings from such psychological studies, we examine the intensity and the duration of a smile in the hope of finding a distinction between the two sexes. Hence, We present an algorithm to measure gender solely based on the dynamics of the smile without resorting to appearance-based image analysis techniques. The dynamic framework we have developed for smile analysis has four key components. They are Face detection, Landmark Detection, Geometric Distances and finally the most important Machine Learning Classification Algorithms

# 2-Related Works

In this Section, we discuss recent advances in face analysis for both smiles and gender classification. Research in these areas appears to be dominantly arising from psychological studies as well as computer aided analysis of both static and dynamic digital images.

In many psychological experiments, the use of facial electromyography (EMG) is common, especially for studies relating to the analysis of the face. EMG is a diagnostic technique used for recording facial muscle activity by placing electrodes on the face . Much work on face analysis have been undertaken using EMG. These include the study of facial reactions to auditory stimuli, gender differences in facial reactions to facial expressions and facial and emotional reactions to both genuine and induced smiles

From a computational viewpoint, gender classification based on the analysis of the face can be divided into three main categories, namely geometric, appearance and methods comprising of a hybrid between geometric and appearance models. All these methods rely on some form of a technique for extracting features from facial images.

The geometric model relies on the spatial geometry derived from a set of facial landmarks for extracting the relevant facial features. These often include physical measurements such as the size of eyebrows, eyes, nose and the mouth. Furthermore, in the dynamic case, this model relies on measures based on the movement of the landmarks.

It has been hypothesized and evidenced by various psychological experiments that there exist differences in smiles between the two genders. To verify this computationally and at the same time to develop a tool for gender classification solely based on the smiles, we propose a framework which can track the dynamic variations in the face from neutral to the peak of a smile. Our framework is based upon four key components.

Face Detection

Landmark Detection

Geometric Distances

Machine Learning Classification Algorithm

Above Figure presents a block diagram showing the key components of our framework for the analysis of the dynamics of a smile. The first step in our framework is to detect and track the face .The next step in our proposed framework is to detect landmarks. After that we have to calculate geometric Distances of face and than we have to use classifiers such as Random Forest , Naives Bayes and Support Vector Machine to detect gender of human .

# 3-Methods

As stated before, we want to encode gender through human smile so we will test this approach on three classifiers which are Random Forest , Naives Bayes and Support vector Machine.

## Dataset description

## In 2000, the Cohn-Kanade (CK) database was released for the purpose of promoting research into automatically detecting individual facial expressions. Since then, the CK database has become one of the most widely used test-beds for algorithm development and evaluation. During this period, three limitations have become apparent: 1) While AU codes are well validated, emotion labels are not, as they refer to what was requested rather than what was actually performed, 2) The lack of a common performance metric against which to evaluate new algorithms, and 3) Standard protocols for common databases have not emerged. As a consequence, the CK database has been used for both AU and emotion detection (even though labels for the latter have not been validated), comparison with benchmark algorithms is missing, and use of random subsets of the original database makes meta-analyses difficult. To address these and other concerns, we present the Extended Cohn-Kanade (CK+) database.

## We have used Ck+ dataset for testing of this approach. We used 60 images which consist of 30 males and 30 females in which smile of a subject is clearly visible.We extract only images which contain the relevant part of the smile of each subject (Happy) from the dataset , So we can easily test our approach .

## Machine Learning strategy

**Random Forest :**

Random forest is a [supervised learning algorithm](https://builtin.com/data-science/supervised-learning-python). The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result.

**Put simply: random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.**

One big advantage of random forest is that it can be used for both classification and regression problems, which form the majority of current machine learning systems. Let's look at random forest in classification, since classification is sometimes considered the building block of machine learning.

Random forest has nearly the same hyperparameters as a decision tree or a bagging classifier. Fortunately, there's no need to combine a decision tree with a bagging classifier because you can easily use the classifier-class of random forest. With random forest, you can also deal with regression tasks by using the algorithm's regressor.

[View 2641 Jobs](https://builtin.com/jobs/data-analytics)

Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.

Therefore, in random forest, only a random subset of the features is taken into consideration by the algorithm for splitting a node. You can even make trees more random by additionally using random thresholds for each feature rather than searching for the best possible thresholds (like a normal decision tree does).

**Naïve Bayes :**

It is a [classification technique](https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2/?utm_source=blog&utm_medium=6stepsnaivebayesarticle) based on Bayes’ Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as ‘Naive’.

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:

[](https://www.analyticsvidhya.com/wp-content/uploads/2015/09/Bayes_rule-300x172.png)

Above,

* *P*(*c|x*) is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
* *P*(*c*) is the prior probability of *class*.
* *P*(*x|c*) is the likelihood which is the probability of *predictor* given *class*.
* *P*(*x*) is the prior probability of *predictor*.

**Support Vector Machine:**

Support Vector Machine (SVM) is a supervised [machine learning algorithm](https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2?utm_source=blog&utm_medium=understandingsupportvectormachinearticle) which can be used for both classification or regression challenges. However,  it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well .



Support Vectors are simply the co-ordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper-plane/ line).

In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM  algorithm has a technique called the [**kernel**](https://en.wikipedia.org/wiki/Kernel_method)**trick**. The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e. it converts not separable problem to separable problem. It is mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you’ve defined .

## Parameters:

|  |  |
| --- | --- |
| Algorithms | Parameters |
| Random Forest | n\_estimators =50 ,  random\_state= 0 |
| Naïve Bayes | - |
| Support vector Machine | C=1.0, cache\_size=200, coef0=0.0,decision\_function\_shape='ovr', degree=3, gamma='auto\_deprecated',kernel='linear',  max\_iter=-1,  shrinking=True, tol=0.001 |

***3.4 Data Preprocessing:***

**Standardization :**

We have used Data Preprocessing technique which is called **Standardization .**Standardization of datasets is a commonrequirement for many machine learning estimators implemented in scikit-learn; they might behave badly if the individual features do not more or less look like standard normally distributed data .To Scale Our data we have used Scikit- learn`s class called **StandardScaler** .

# 4 - Results

For each classifier has been carried out several tests, and in the table [w](#_bookmark3)e show the best-obtained results according to accuracy.

***TEST SIZE = 30 %***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Classifiers*** | ***Accuracy*** | ***Precision*** | ***Recall*** | ***F1 Score*** | ***iteration*** |
| Random Forest | 55% | 57% | 44% | 50% | 0 |
| Naives Bayes | 44% | 75% | 25% | 38% | 0 |
| Support Vector Machine | 55%  61% | 50%  71% | 38%  50% | 43%  59% | 0  1 |

***TEST SIZE = 20 %***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Classifiers*** | ***Accuracy*** | ***Precision*** | ***Recall*** | ***F1 Score*** | ***iteration*** |
| Random Forest | 83% | 67% | 100% | 80% | 0 |
| Naives Bayes | 75% | 100% | 50% | 67% | 0 |
| Support Vector Machine | 66% | 67% | 67% | 67% | 0 |

**5- CONCLUSION :**

In this project we work on three different classifiers they are Random Forest , Naives Bayes and Support Vector Machine(SVM).Our best accuracy on test size 30% is 61% in second Iteration of SVM and on test size 20% Random forest gives us best accuracy of 83% .So in our case SVM Gave us better accuracy than others while when we reduce our test size than random forest gave us better accuracy .