# A New Approach of Facial Expression Recognition for Ambient Assisted Living

Yacine Yaddaden, Abdenour Bouzouane, Mehdi Adda, Bruno Bouchard

LIARA Laboratory
University of Quebec at Chicoutimi (UQAC)
555, Boulevard de l'Université
Chicoutimi (Quebec), Canada, G7H 2B1
{yacine.yaddaden1, abdenour_bouzouane, bruno_bouchard}@uqac.ca,
{mehdi_adda}@uqar.ca

## ABSTRACT

Ambient Assisted Living and Ambient Intelligence have seen their impact greatly grows, especially these last decades. It is mainly due to the increase of the ageing population and people with cognitive diseases. Several technologies were developed to make the use of assistive technology more acceptable and comfortable for the elderly in order to reduce or even replace the human assistance. However, there are many challenges and issues, especially in the interaction between the elderly and assistive systems. To make the system interact as human beings, emotions were used. In this paper, we present a new approach to recognize emotions based on facial expressions represented by images. It is based on a new method for feature selection based on distances. We also suggest the use of the well-known K-Nearest Neighbor classifier with optimized parameters. This approach is found effective when tested using two different datasets of images.

## CCS Concepts

•**Emotion recognition** → **Ambient Assisted living;**

## Keywords

Emotion recognition, facial expressions, data mining, ambient intelligence, ambient assisted living

## 1. INTRODUCTION

In order to meet the daily needs of the elderly with cognitive diseases, several initiatives have been undertaken and several technologies have been proposed [17]. The smart homes initiative is the most used [21]. It consists of an intelligent environment which contains sensors to collect information and actuators to trigger actions. However, a few issues were encountered, because the targets are the elderly and they are not familiar and comfortable with technology. Probably because technological devices look complicated and they are afraid of doing something wrong.

Researchers proposed the use of social entities to address this issue. They act as an intelligent intermediary between the smart home and the elderly. Robots were proposed to collect information from different sensors and services in the smart home and interact with the elderly in order to make the use of technology more acceptable [14]. However, for an efficient interaction between the robots and the elderly, the robots have to match the human model in terms of interaction and communication. To achieve this goal, the emotion is the most useful and accurate modality. It is the most universal communication method for human beings. This is why, emotion recognition is gaining widespread interest among researchers in the field of AAL (Ambient Assisted Living) and AmI (Ambient Intelligence) [18].

The most common way used by humans to express emotions and communicate is the facial expression. Mehrabian [15] estimates that facial expressions contribute greatly to the effect of the message by 55%, while the verbal and vocal parts contribute respectively with 7% and 38%. There are two different sources for emotion recognition from facial expressions: 2D signal (Images Based) and 3D signal (Sequences or Videos Based). Working with video has as advantage the huge amount of information introduced by the temporal aspect. However, video-related operations are generally known to be CPU/GPU and RAM intensive tasks. Contrariwise, images hold smaller amount of information. Thus, the related operations (such as activity recognition tasks) require fewer resources. It makes them more suitable for embedded systems with limited resources. Paul Ekman [4] presented many research works and studies in this field and proposed to divide emotions into a set of six basic emotions: fear, happiness, sadness, disgust, surprise and anger. He also proposed a specific method (Facial Action Coding System (FACS)) to detect these emotions from facial expressions. This method can be applied only on a sequence of images; it is based on the movement of specific face muscles which means that the temporal aspect is required.

Emotion recognition was always considered as challenging, especially from images because of the small amount of available information. There are other issues and limitations related to the acquisition conditions: scale, luminance, orientation, obstruction, etc. The feature extraction and se-



*PETRA '16, June 29-July 01, 2016, Corfu Island, Greece*



DOI: http://dx.doi.org/10.1145/2910674.2910703

lection are also challenging parts because of the existence of different techniques which make the choice of the appropriate method very difficult. But the most important and critical part of the process of emotion recognition is the classification step. The challenge here is to choose a method, among all existing ones, which is less resource hungry and in the same time be as accurate as possible. It is well-known that the classification method depends on the used features. Hence, a comparative study must be performed with a specific type of feature in order to choose the most adequate and efficient classification method.

In this paper, we introduce a new approach for emotion recognition based on facial expressions contained in images. The used features in order to characterize each facial expression consist of the distances between the FCP (Facial Characteristic Points) or Landmarks of the face. Other authors [5] and [23] have already used the same kind of features. However, the main difference is the choice of the distance. In their works, the used distances were chosen subjectively or based on psychological experimentations. This approach produced interesting results, but the number of features is limited and not weighed by importance. In our approach, the distances are chosen based on a statistical measure which makes our method more objective, automatic and adaptive. For the classification step, we suggest the use of K-Nearest Neighbors (KNN) classifier for its simplicity and easy implementation. However, this method has several parameters which must be optimized to get the best performances in terms of accuracy and efficiency. In order to validate the proposed approach, we used two different datasets to compare our method to existing ones.

This paper is organized as follows. After beginning with an introduction to summarize our work, some previous and related works will be presented in the section II. Afterwards, all the details of the proposed method will be presented in the section III. Then, the results of the experimentations will be discussed in the section IV. Finally, a conclusion about the present work and perspectives will be presented.

## 2. RELATED WORKS

Emotion recognition can be described as a classical pattern recognition system [7]. The input data can be a 2D signal (Images) or 3D signal (Videos or a Sequence of images). The first step consists of features extraction. There are three types of features [23]: 1) Geometric which uses an appropriate model like Active Shape Model (ASM), 2) Appearance which exploits the entire image at pixel level. Some transformations like Gabor Wavelets can be applied to the image and 3) Hybrid which combines the two previous approaches. The next step focuses on reducing the features vector by selecting only the most representative and pertinent ones. Several methods exist based on Principal Component Analysis (PCA). The final step of the process is the classification. It is based on a supervised machine learning method which means that it requires a training data in order to do predict unknown or unlabeled data. In the following, several approaches will be presented with different methods in each one of the steps of the facial expression recognition process.

Hai et al. [5] proposed a hybrid approach for emotion recognition based on facial expressions. Their contributions focus on the feature extraction and classification parts. They used two different features, the first one is based on the coefficients generated from the ICA (Independent Component Analysis) and the second one consists of ratios computed by using different distances. For the classification step, for each kind of feature, they used a different classification method. For the first type of feature (ICA coefficients), ANN (Artificial Neural Network) classifier was used and for the second type of feature (Distance ratios), KNN classifier has been used. Their method has permitted to achieve, with the JAFFE dataset, the following classification rates: 91.43% with ICA/ANN, 90.48% with KNN/Distance and 92.38% by combining the two methods. Their approach provides pretty interesting results but they used two different feature extraction methods which increase the complexity of their system, computation time and resource consumption. The combination of both methods did not improve greatly the accuracy and the ratios of distances used as features were defined subjectively.

Li et al. [10] proposed a method which uses the appearance based features. The feature extraction step is divided into two parts. The first one is the application of fixed filters in order to extract primitive features which consist in a set of coefficients. Then, adaptive filters are applied in order to extract more complex features. After that, the most relevant and representative features are selected and extracted. Finally, the classification method which is based on SVM (Support Vector Machine) is applied. With these methods, the authors achieved 96.7% classification rate when performed on the JAFFE dataset. Even if the accuracy of this approach is pretty good, but the feature extraction method is pretty complex and generates a very high number of coefficients which can impact the computation time and resource consumption. This method is not adapted to an embedded system with limited resources such as a robot.

Lajevardi and Hussain [8] introduced a new method for feature extraction. It is based on HLAC (Higher-Order Local Auto Correlation) and LBP (Local Binary Pattern) [2]. The first step of their proposed method is to extract the features and then apply a method based on MIC (Mutual Information Quotient) in order to select the most pertinent features. The selected and validated features are used as input to the classification process which is based on the NB (Naive Bayes) classifier. Their method achieved 65.5% and 69% classification rate respectively with HLAC and LBP. The experimentations were performed on the Cohn-Kanade dataset. The authors observed that when the resolution of the input images increase, HLAC based features give better results than LBP. This is a new approach based on appearance-based features even if the LBP was already used in facial expression recognition. It is pretty complex and can be difficult to implement in an embedded system. The obtained results in terms of accuracy are low and insufficient.

Youssif and Asker [23] used to combine in their proposed method the appearance and geometric based features to recognize emotions from the facial expressions. The first step consists of extracting features based on the face zone, normalizing the result image, dividing the face zone into three different zones (mouth, nose, eyes-eyebrows), generating the FCP or Landmarks and extracting the geometric based features. The appearance-based features consist of the coefficients extracted from the application of the Canny Edge Detector on the face image. These two kinds of features are combined to form a unique feature vector. In the classification phase, the ANN classifier with RBF (Radial Basis

Function) was used. The experimentations were performed on the Cohn-Kanade dataset and they achieved 93% classification rate. The accuracy of the proposed system is good. However, the combination of two different approach increase the complexity and the resource consumption of the system.

Sohail and Bhattacharya [19] proposed a method of facial expressions recognition based on geometric features. They consist of eleven Euclidean distances. Before generating these distances, the authors performed some image processing in order to extract the FCP or Landmarks. The method begins by detecting the eyes. Based on their position, they detect the other zones of interest: eyebrows, mouth and nose. At this stage, all the zones of interests are limited by rectangle boundaries. Based on that and for each zone, they generate the FCP or Landmarks. From these FCP, they generate the eleven Euclidean distances in order to constitute the feature vector. The used classification method is the KNN with $K = 3$. When it was applied on the JAFFE dataset, they obtained 90.76% classification rate. This result is the best compared to the other methods: 83.19% with ANN and 84.05% with NB.

There are other recent approaches which are based on Deep Learning [9].Unlike classical methods where the features are manually defined, Deep Learning approaches exploit the hierarchical representations of the input images. The representations are extracted automatically by applying different filters. The most used Deep Learning architecture in vision is the CNN (Convolutional Neural Network) which consists of several layers. In each layer, a certain level of feature is extracted. It is a supervised method which means that the dataset is divided into two different subsets. The first one is used for the training phase with the back propagation method. The second one is used to validate the generated model. Lv et al. [12] divided their process of emotion recognition from facial expressions into two subprocesses. The first one uses the DBN (Deep Belief Network) in order to detect the different components of the face. The second one takes as input the different component and performs classification with SA (Stacked Autoencoder). The obtained results with JAFFE dataset is 90.74% and 91.11% with Cohn-Kanade extended dataset. Song et al. [20] have developed a smartphone application with a client-server architecture. They exploit a specific architecture of CNN. They obtained interesting results with Cohn-Kanade extended dataset (99.3%). In order to overcome the issues related to overfittig, the authors have used different methods. One of them consists of increasing the size of the image dataset by performing some transformations e.g. mirroring. Another method is directly linked to the network; it is called **Dropout**. Basically, it consists of ignoring some nodes of a hidden layer with a certain probability. This is done during the training phase.

# 3. PROPOSED METHOD

Taking into account the weaknesses of the studied approaches, we propose a new one which overcomes these limitations. The process of this method is presented in Figure 1. It is adapted for a real-time use and implementation on embedded systems. From a dataset of facial expression images, a feature extraction is applied. It is divided into two sub steps: FCP or Landmarks extraction and calculation of distances. For the feature selection step, we propose a new method based on a statistical measure. It has a parameter which represents the threshold. Last, but not least, for the classification step, we propose the use of KNN for its simplicity, fast speed and easy to implement. We will also present the optimized parameters for the used KNN algorithm in order to get a higher accuracy as possible. Depending on the value of the obtained accuracy, the threshold is updated. The list of selected features is generated when the value of the accuracy converges.

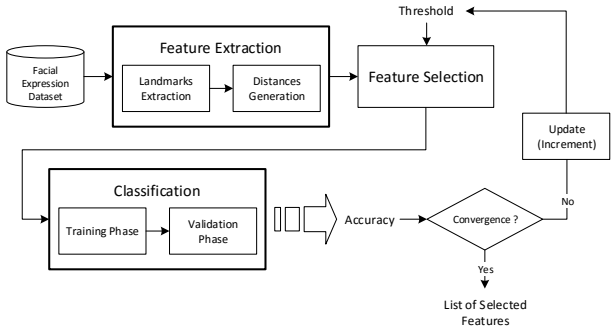


**Figure 1: Process of the proposed method**

## 3.1 Feature Extraction

The used features consist of a set of distances. Before that, the FCP or Landmarks must be extracted from the face image. We propose the use of a method which was introduced by Kazemi and Sullivan [6] and it consists of a cascade of regression functions. The first one takes as input the initial estimate of the face shape and the intensities of sparse set of pixels indexed relative to the initial estimate. It generates a new adjusted and rectified estimate. The same process is done in the next regression functions until convergence.

We suppose $P_i \in \mathbb{R}^2$ is the coordinate $(x, y)$ of the $i$th landmark or FCP extracted from the estimate of a face shape generated from an input image $I$. The vector $S = \{P_1, P_2, ..., P_N\} \in \mathbb{R}^{2N}$ contains the coordinates of all the Landmarks. There is an ensemble of regression functions, each one is noted $r_t()$ and has two different arguments. The first one is the current estimate of face shape $\hat{S}^t$ and the second one is the image $I$. The output of the next regression function is defined by the expression:

$$\hat{S}^{t+1} = \hat{S}^t + r_t(I, \hat{S}^t) \quad (1)$$

For the first and initial face shape, the mean of all the face shapes of the training data can be used. It must be centered and rescaled to match the bounding box of the face detector. The method exploited to detect the face was introduced by Viola and Jones [22]. It is the most common method because of its high accuracy, low consumption of resources and fast speed.

Each regression function is represented by a decision tree. The ensemble or cascade trees are trained by using the gradient boosting approach. A specific weight is associated with each regression function output according to its accuracy or error rate. If the error rate is low, the value of the weight is high. The method is iterative until convergence to the appropriate and perfect estimate of face shape. It is adapted to a real-time use, the training phase takes approximately one

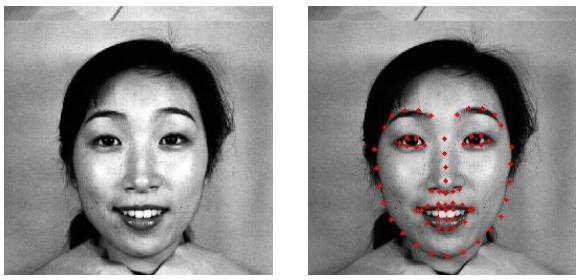

**Figure 2: (a) Original Image (b) Image with FCP**

hour and the predicting phase takes a millisecond. It was compared to other existing methods: *STASM* and *CompASM* based on ASM (Active Shape Model) and it was proved that this method provides better results.

After extracting all the FCP or Landmarks from the input image, the next step is to generate all possible Euclidean distances. If we suppose that we have two different FCP $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$, the Euclidean distance between these two FCP:

$$D(P_1, P_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2)$$

There are $N = |S| = 68$ different extracted FCP. The number of all generated distances $M$ is given by the following expression:

$$M = \sum_{i=1}^{N} N - i \quad (3)$$

According to the number of extracted FCP, the number of possible distances is $M = 2278$. There are many distances which represent the feature of our system of emotion recognition from facial expressions. It must be reduced.

## 3.2 Feature Selection

The features are not all representative or pertinent. There is only a subset of features which is useful. The other ones can introduce errors or increase the computation time and the amount of consumed resources. For this reason and in order to reduce the size of the feature vector, we will use a criterion which is based on a statistical measure, the variance:

$$\sigma_i^2 = \frac{1}{7} \sum_{j=1}^{7} (D_j^i - \bar{D}^i)^2 \quad (4)$$

$\sigma_i^2$ corresponds to the variance of the $i$th distance. $i$ varies in $[1, M = 2278]$ corresponding to all possible Euclidean distances. $j$ in the expression 4 is represented in $[1, 7]$ which correspond to the seven emotions.

The datasets which will be used are organized by persons and each one can be defined as a matrix. Each row of the matrix represents a specific emotion: fear, happiness, sadness, disgust, surprise, anger and neutral. Each column represents a specific distance or feature. The process of selecting the most representative features can be divided into two different sub steps:

In the first step, for each person in the dataset (see line 2), the different variances corresponding to each distance are computed (see line 4). This operation is performed on each column (see line 3). After that, the distances are ordered decreasingly depending on the value of their variances (see line 7). In order to select only some distances, a threshold $T$ is fixed and applied. According to the value of the threshold, the distances with the higher variances are extracted (see line 8). This operation is repeated along all the dataset. All the distances are stored in a unique vector Y. Then, all the occurrences of the different distances are computed and ordered decreasingly (see line 11 and 12). Finally, the threshold $T$ is applied to get the final selected distances (see line 13 and 14).

**Algorithm 1:** Feature Selection

```
   Data: Dataset of FCP, Threshold T
   Result: Vector of Selected Distances Y
 1 begin
 2     foreach Subject in Dataset do
 3         foreach Distance D^i do
 4             Compute the variance σ_i^2
 5             Store σ_i^2 in the vector X
 6         end
 7         Order X decreasingly
 8         Apply threshold T to the vector X
 9         Add the vector X to the vector Y
10     end
11     Compute the occurrence of each element of Y
12     Order elements of Y decreasingly by occurrence
13     Apply threshold T to the vector Y
14     return Resulting vector Y
15 end
```

The most representative features are the ones which vary the most when the person expresses different emotions. The variance $\sigma_i^2$ represents the changes and variations of the value of the different feature when the facial expression changes. For this reason, it has been used as the main criterion for feature selection. It will be proven in the results that, for the most used classifiers, the accuracy increases following the increase of the threshold $T$ until convergence. This means that the variance is an effective criterion to use.

$T$ is a critical parameter because the number of retained features or distance depends on it. Its value is fixed automatically by our system during the validation phase (see Figure 1). The value depends on the accuracy and when the high classification rate is reached, the retained threshold corresponds to it.

## 3.3 Classification Algorithm

KNN is an instance based learning method. This concept was introduced for the first time by Aha et al. [1]. It consists of a set of example data which are stored and accessible. There is no step of generating models. This is why, this kind of method is commonly called lazy compared to the other methods called eager which generate a model. In order to predict an unknown or unlabeled data, a certain number of the closest neighbors are extracted by using a distance measure. The label of the unknown data is defined by a majority vote between the labels of the closest neighbors. This classification method has different parameters which must be defined in order to construct the most efficient KNN based classifier.

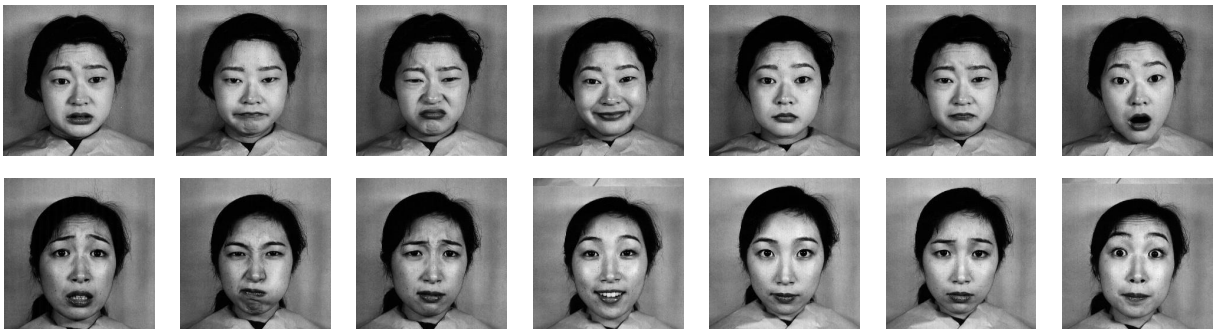

**Figure 3: Two subjects from JAFFE dataset expressing the seven emotions**

The first parameter is the number of the closest neighbors which will be used in the prediction phase. The second one which is the most important is the distance measure. There are a lot of distance measure, the most common and used are: Euclidean and Manhattan. However, in this paper, we will use two different distance measures which are more adapted to this kind of feature: Cosine and Correlation. These two distances were chosen because of the high classification rate they achieve. In the most of works with KNN, the used distance is the Euclidean. But in our case, it does not permit to reach the best results. This is why, we tried different distance until we found the Cosine and the Correlation which are more adapted to this kind of feature.

$$D_{Euclidean} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{5}$$

$$D_{Manhattan} = \sum_{i=1}^{n}|x_i - y_i| \tag{6}$$

$$D_{Cosine} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \times \sqrt{\sum_{i=1}^{n} y_i^2}} \tag{7}$$

$$D_{Correlation} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \times \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{8}$$

KNN has a drawback which is the increase of computation time and the amount of consumed resources if the training data comes to increase **vertically** (Size of the dataset) or **horizontally** (Number of features or attributes). This method can be implemented in two different ways. The unstructured approach which is the most common way, also known as "Brute Force". In order to classify an unknown data, the system must browse all the dataset in order to find the closest neighbors. The structured approach is more complex but reduces greatly the computation time. The principle consists of organizing the training data in a hierarchical way and generate a tree. This operation is performed once and reduce the time of browsing. There are two common implementations of this approach. KD-Tree, it was introduced by Bentley in 1975 [3] which make a subdivision of the training data in form of hyper-rectangles of a specific dimension. The second one is the Ball-Tree which was introduced by Omohundro in 1989 [16]. It consists of a subdivision of the training data in form of hyper-spheres of a certain dimension.

These two hierarchical and structured methods will help to reduce the computation time in the research of the closest neighbors. A benchmark between these different methods will be presented to show how efficient they are when applied in the facial expression recognition.

# 4. EXPERIMENTATION

In this section, we will present our experimentation protocol, performed experiments, obtained results and an analysis of those results.

## 4.1 Datasets

For the needs of the experimentations, two different datasets will be used. They contain images of persons expressing emotions which are in number of 7: fear, happiness, sadness, disgust, surprise, anger and neutral.

The first dataset is the JAFFE (Japanese Female Facial Expression) [13]. It contains 213 gray level images of seven facial expressions (six basic facial expressions and neutral) of 10 Japanese female models. The dataset is pretty old (1999) and there are not enough models. The quality of the images is not very good but since it is old, it is the most common and used dataset in the field of facial expression and emotion recognition.

The second dataset, KDEF (Karolinska Directed Emotional Faces) [11] has a better quality of the images. It contains 4900 color images of seven facial expressions (six basic facial expressions and neutral) with five different angles. 70 subjects have participated to constitute the dataset. Our motivation for using this dataset is due to its size and the ethnic origins of the models because the JAFFE dataset contains only Japanese and female models. In this dataset, there are female and male models and they are all Europeans.

In Figure 3 is represented two different subjects from the JAFFE dataset. They express the 7 emotions, the 6 basics defined by Paul Ekman and the neutral state. It was preferable to use datasets of persons with cognitive impairment, but we did not find ones. Though, our method can be easily applied to other datasets.

## 4.2 Results and Discussion

In our work, we performed different experimentations. Each one has generated some interesting results. The clas-

sification method we proposed is the KNN but it has different parameters which are crucial and the results are closely linked to these parameters. The first parameter is the distance measure which will be used in order to extract the closest neighbors. For our experiment, we have compared four different distances; among them, two are very common and widely used: Euclidean and Manhattan distance. We proposed the use of two other distances which are: Cosine and Correlation distance.

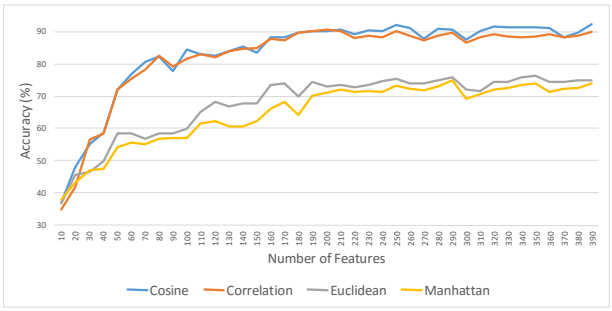


**Figure 4: KNN Distance Benchmark - JAFFE**

From Figure 4 and 5, we can see that the most efficient distances are the Cosine and Correlation distances compared to the ones of Euclidean and Manhattan distances. The best classification rate achieved with the Cosine and Correlation distances are respectively 92.29% and 90.76% with the JAFFE dataset, 79.69% and 79.80% with the KDEF dataset. With the other distances which are the Euclidean and Manhattan, the best-achieved classification rates are respectively 76.40% and 74.83% with the JAFFE dataset, 71.73% and 71.63% with the KDEF dataset.

The obtained results show that the two most efficient distance measures for this kind of features are the Cosine and Correlation distances. For the next set of experiments, we selected the Cosine distance because it gives the best results.

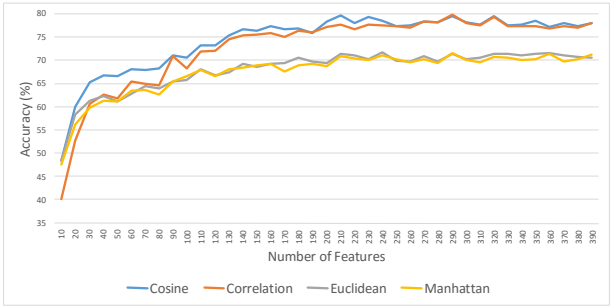


**Figure 5: KNN Distance Benchmark - KDEF**

The second parameter is the number of closest neighbors $K$. It is a critical parameter and, the classification rate is closely linked to it. The $K$ parameter which stands for the number of closest neighbors can be found by iterative tests. There is another alternative to the classical KNN which is the Weighted K-Nearest Neighbors (WKNN). The principle of this method is simple, it is exactly similar to the classical KNN but for each closest neighbor, it assigns a weight which depends on the distance between the unknown or unlabeled data and the closest neighbors. In Figure 6, we can see that the best results are achieved with $K = 1$ in both methods: classical KNN and WKNN. When we increase the value of $K$, the WKNN shows better performances than KNN.

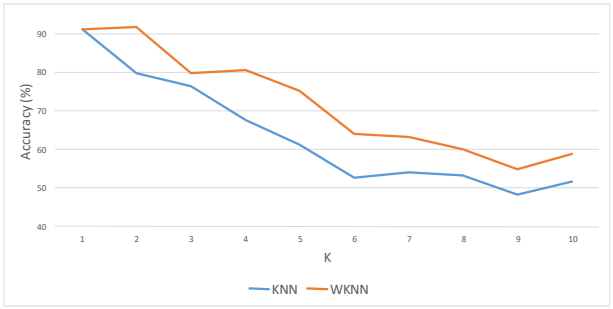


**Figure 6: KNN compared to WKNN**

For now, we have defined the best parameters to use with the KNN method in order to achieve the best performances. The used distance measure is the Cosine distance with the number of closest neighbors $K = 1$. We discussed the main drawback of the KNN which is the computation time in the browsing of the dataset in order to find the closest neighbors and mostly when the dataset is huge **horizontally** (Number of features or attributes) or **vertically** (Size of the dataset). There are two different hierarchical and structured methods: KD-Tree and Ball-Tree which reduce greatly the computation time. In Figure 7, we compare the two different methods with the classical one which is the Brute Force approach.

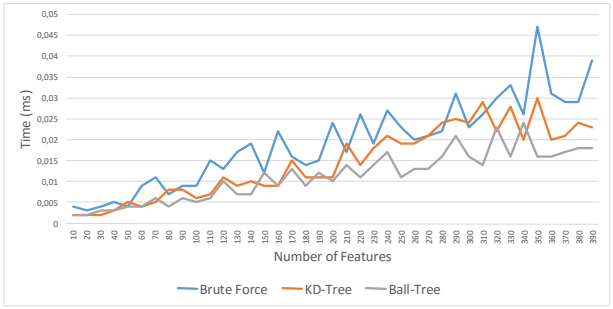


**Figure 7: Structured and Unstructured approaches**

The obtained results show that the best performances are achieved with the structured approaches. The best hierarchical method which reduces the computation time is the Ball-Tree compared to the KD-Tree and the Brute Force approach.

In the Figure 8 and 9, we applied the different classification methods: KNN, ANN, DT (Decision Tree) and SVM on the two different datasets JAFFE and KDEF. The input of each classifier is the selected distances which represent the features of each one of the two different datasets. The two most efficient classification methods are the KNN and the ANN which give respectively the classification rate of

**Table 1: Comparison with other methods**

| Methods | Features | Algorithms | Datasets | Accuracy |
|---|---|---|---|---|
| Hai et al. [5] | Appearence | ANN | JAFFE | 91.43% |
| | Geometric | KNN | | **90.48%** |
| | Hybrid | ANN + KNN | | 93.00% |
| Youssif et al. [23] | Hybrid | ANN | Cohn-Kanade | 93.00% |
| Li et al. [10] | Appearence | SVM | JAFFE | **96.70%** |
| Lajevardi et al. [8] | Appearence (LBP) | NB | Cohn-Kanade | 96.00% |
| | Appearence (HLAC) | | | 65.50% |
| Sohail and Bhattacharya [19] | Geometric | KNN | JAFFE | **90.76%** |
| | | ANN | | 83.19% |
| | | NB | | 84.05% |
| Lv et al. [12] | Face Components | SA | JAFFE | 90.74% |
| | | | Cohn-Kanade+ | 91.11% |
| Song et al. [20] | CNN | | Cohn-Kanade+ | 99.30% |
| **Our Method** | Geometric | KNN | JAFFE | **92.29%** |
| | | | KDEF | 79.69% |

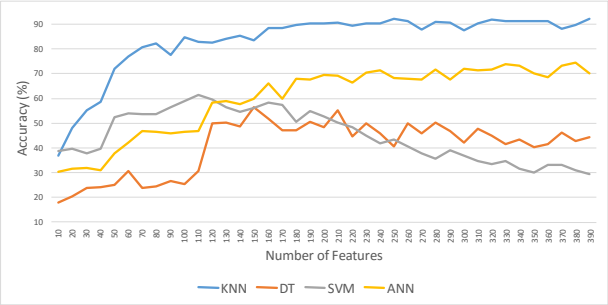


**Figure 8: Comparison between algorithms - JAFFE**

92.29% and 74.48% with the JAFFE dataset.

In the KDEF dataset, the best performances are also achieved with the KNN and the ANN which achieved respectively the classification rate of 79.69% and 79.90%.

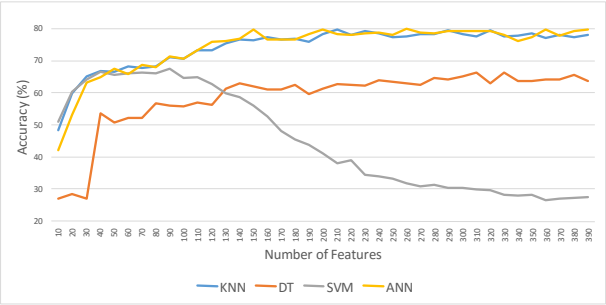


**Figure 9: Comparison between algorithms - KDEF**

There is a difference between the KNN and ANN. The second one is too complex and difficult to implement and the computation time for the training phase is very high compared to the other methods as shown in Table 2.

The computation time for the training phase is very important especially if the aim is an implementation on an embedded system with limited resources. The KNN is perfect for this application because of its simplicity and the less computation time. From Table 2, we can see that for the KNN, the computation time represents only the storing of the data, this is why it has the lowest computation time in the training phase compared to the other classification methods.

**Table 2: Computation time - Training Phase (ms)**

| Datasets | KNN | DT | SVM | ANN |
|---|---|---|---|---|
| *JAFFE* | **2** | 230 | 65 | **119718** |
| *KDEF* | **10** | 868 | 1402 | **557481** |

From the Table 1, we can see that the best accuracy is reached with the method proposed by Li et al. [10] with 96.70% classification rate. But this method has some drawbacks like the use of SVM classifiers which is considered as black box because of its complexity and it is pretty hard to implement on an embedded system. The kind of used features which are appearance based can increase the computation time and resource consumption, especially in an embedded system with limited resources. The Deep Learning approach gives very interesting results but there are some limitations related to the resources, especially for the training phase and also the size of the dataset must be appropriate in order to avoid overfitting. The accuracy we reached, when applying our method, is lower ($\approx$ 4%) but it is simpler and easier to implement in terms of the chosen classifier. The geometric based feature we use is extracted and selected with a new approach which is less resource hungry. Also, the computation time and the number of used features are much lower (Number of Features $\approx$ 310). The other methods proposed by Hai et al. [5] and Sohail and Bhattacharya [19] can be compared with our method because they used the same kind of features, the same classifier and we achieved better results. The features they used are selected subjectively in contrast of our method which is based on a statistical measure. It makes our method more objective. The other approaches were tested on a different dataset and this is why we cannot make an objective comparison.

## 5. CONCLUSION

In the present work, we presented a new approach of emotion recognition based on facial expressions from images. We have obtained interesting results in comparison with other previous works. It represents the first step of our work which targets the use of emotion in the fields of AAL and AmI. It has not only for objective to enhance the interaction between the elderly and the assistive technologies but it also aims to detect the committed errors. This application will be based on the facial expression; it will be an automatic approach to detect the errors. However, our method needs to be improved in its accuracy and also be extended to 3D signals (Videos). This extension will be permitting better real-time applications. The Deep Learning approach seems to be very promising. It worth being exploited in the field of emotion recognition from facial expressions.

## 6. ACKNOWLEDGMENTS

We would like to acknowledge our main financial sponsors: UQAC (University of Quebec at Chicoutimi) and UQAR (University of Quebec at Rimouski). We also would like to acknowledge the providers of the two datasets of facial expression images: The JAFFE dataset [13] and the KDEF [11].

## 7. REFERENCES

[1] D. W. Aha, D. F. Kibler, and M. K. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[2] T. Ahonen, A. Hadid, and M. Pietikäinen. Face recognition with local binary patterns. In *Computer Vision - ECCV 2004, 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, PartI*, pages 469–481, 2004.

[3] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, 1975.

[4] P. Ekman. An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200, May 1992.

[5] T. S. Hai, L. H. Thai, and N. T. Thuy. Facial expression classification using artificial neural network and k-nearest neighbor. *International Journal of Information Technology and Computer Science*, 7(3):27–32, feb 2015.

[6] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *CVPR*, pages 1867–1874. IEEE, 2014.

[7] A. Konar, A. Halder, and A. Chakraborty. Introduction to emotion recognition. In *A Pattern Analysis Approach*, pages 1–45. Wiley-Blackwell, jan 2015.

[8] S. M. Lajevardi and Z. M. Hussain. Local feature extraction methods for facial expression recognition. In *17th European Signal Processing Conference, EUSIPCO 2009, Glasgow, Scotland, UK, August 24-28, 2009*, pages 60–64, 2009.

[9] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, may 2015.

[10] P. Li, S. L. Phung, A. Bouzerdoum, and F. H. C. Tivive. Feature selection for facial expression recognition. In *2nd European Workshop on Visual Information Processing, EUVIP 2010, Paris, France, July 5-7, 2010*, pages 35–40, 2010.

[11] D. Lundqvist, A. Flykt, and A. Öhman. The karolinska directed emotional faces - kdef, cd rom from department of clinical neuroscience, psychology section, karolinska institutet, 1998.

[12] Y. Lv, Z. Feng, and C. Xu. Facial expression recognition via deep learning. In *Smart Computing (SMARTCOMP), 2014 International Conference on*, pages 303–308, Nov 2014.

[13] M. J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. Coding facial expressions with gabor wavelets. In *FG*, pages 200–205. IEEE Computer Society, 1998.

[14] C. McCREADIE and A. TINKER. The acceptability of assistive technology to older people. *Ageing & Society*, 25:91–110, 1 2005.

[15] A. Mehrabian. *Communication without words*, pages 51–52. 2 edition, 1968.

[16] S. M. Omohundro. Five balltree construction algorithms. Technical Report TR-89-063, International Computer Science Institute, December 1989.

[17] P. Rashidi and A. Mihailidis. A survey on ambient-assisted living tools for older adults. *IEEE J. Biomedical and Health Informatics*, 17(3):579–590, 2013.

[18] P. Saini, B. E. R. de Ruyter, P. Markopoulos, and A. J. N. van Breemen. Benefits of social intelligence in home dialogue systems. In M. F. Costabile and F. Paternò, editors, *INTERACT*, volume 3585 of *Lecture Notes in Computer Science*, pages 510–521. Springer, 2005.

[19] A. S. M. Sohail and P. Bhattacharya. Classification of facial expressions using k-nearest neighbor classifier. In A. Gagalowicz and W. Philips, editors, *MIRAGE*, volume 4418 of *Lecture Notes in Computer Science*, pages 555–566. Springer, 2007.

[20] I. Song, H. J. Kim, and P. B. Jeon. Deep learning for real-time robust facial expression recognition on a smartphone. In *Consumer Electronics (ICCE), 2014 IEEE International Conference on*, pages 564–567, Jan 2014.

[21] M. R. Tomita, L. S. Russ, R. Sridhar, and B. J. N. M. Smart home with healthcare technologies for community-dwelling older adults. In *Smart Home Systems*. InTech, feb 2010.

[22] P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[23] A. A. A. Youssif and W. A. A. Asker. Automatic facial expression recognition system based on geometric and appearance features. *Computer and Information Science*, 4(2):115–124, 2011.