

An Overview: Guided Particle Swarm Optimization (GPSO) Algorithm for facial emotion detection

Suchita N. Sangvikar

Student, Dept. of Electronics Engineering
SPWEC, Aurangabad,
Maharashtra, India

Dr. G. S. Sable

Head, Dept. of Electronics Engineering
SPWEC, Aurangabad,
Maharashtra, India

Abstract—Emotions play a major role in a Human life. These emotions are reflected in voice, hand and body gestures, and mainly through facial expressions. At different kind of moments or time Human face reflects that how he/she feels or in which mood he/she is. This paper presents a novel approach to a modified Particle Swarm Optimization algorithm, which we called Guided Particle Swarm Optimization (GPSO) which can be useful in detection of facial emotion. A guided Particle Swarm Optimization Algorithm which is another PSO algorithm in modified version is proposed in this paper. The approach of the algorithm involves tracking the movements of 10 Action Units (AUs) which are to be placed at appropriate points on the face of a person. Every time two particles are taken in an action unit and by calculating the current position and velocity the This paper will focus on the overview of GPSO and PSO algorithm which can be further implemented actually for the facial emotion detection.

Keywords—guided particle swarm optimization (GPSO), action units (AUs), facial expressions, emotion detection.

I. INTRODUCTION

There are many ways that humans can express their emotions. The most natural way to express emotions is using facial expressions. Humans are capable of producing thousands of facial actions during communication such as happy, sad, angry, fear, surprised, neutral etc. that vary in complexity, intensity, and meaning.

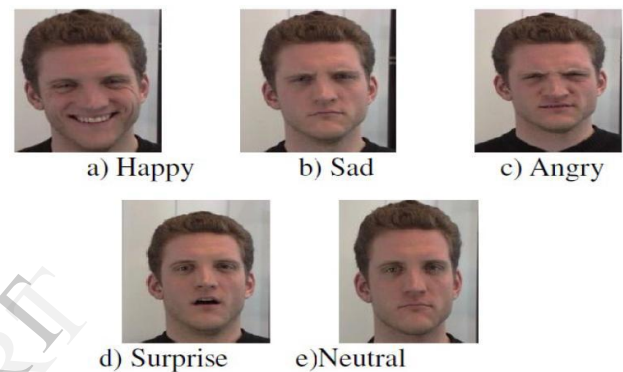


Figure 1: Different Human emotions

Figure 1. is showing different Human emotions which are able to detect using GPSO algorithm. A human can express his/her emotion through lip and eye [17]. A category of emotions which universally developed by Ekmen are: sadness, angry, joy, fear, disgust and surprise without consider natural emotion [1].

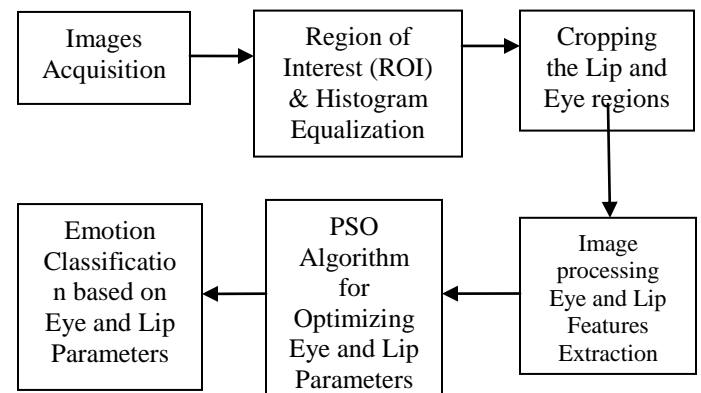


Figure 2: Emotion recognition process

The general process for emotion recognition is shown in Figure 2. The process consists of mainly three parts. The first part describes various stages in image processing include preprocessing, filtering, edge detection and projection profile

which is used to extract features. The second part discusses a PSO based approach to optimize the eye and lip ellipse characteristics. In the third part based on the eye and lip optimal parameters the emotions are classified.

A. Necessity

Facial expressions give important clues about emotions. So, it is necessary to identify the emotion on human's face being expressed at each frame in the video clip or in the picture. Therefore, several approaches have been proposed to classify human affective states. So, in order to fulfill the need of emotion detection the modified guided particle swarm optimization algorithm is proposed in this paper. The features which are used in the PSO algorithm are typically based on local spatial position or displacement of specific points and regions of the face, unlike the approaches based on audio, which use global statistics of the acoustic features.

II. BACKGROUND

Particle swarm optimization (PSO) is a population based, self-adaptive search optimization technique, firstly introduced by Kennedy and Eberhart in 1995 [2]. The motivation for the development of this method was based on the simulation of simplified animal social behaviors such as fish schooling, bird flocking, etc. As PSO is inspired from bird flocking it uses velocity equation to update the solutions and fly towards the best solution.

PSO either in its original form or with some modifications was soon found to be applicable in solving a variety of problems. Examples of its application include the classical travelling salesman problem [3], electrical power systems [4] and neural networks training [5]. It has been applied to clustering problems such as image clustering [6], data clustering [7] and Gene clustering [8]. Other applications of PSO are in the areas of underwater acoustics [9], task assignment [10] and combinational logic circuits design [11], etc. However, to our knowledge, PSO has not been applied directly in solving emotion detection problems.

PSO algorithm has been widely used in various fields [2, 3] due to its few parameters to adjust, easy to understand, easy to implement, and computationally efficient. Despite of these advantages, PSO is easy to be trapped into local optima, and the convergence rate decreased considerably in the later period of evolution processing.

A. Existing Technology

One approach to facial expressions classification is to recognize the underlying facial muscle activities and then interpret these in terms of categories such as emotions, attitudes or moods [13]. The Facial Action Coding System (FACS) [11] is the best known and the most commonly used system developed for human observers to describe facial

activity in terms of visually observable facial muscle actions (i.e., Action Units, AUs). With FACS, human observers uniquely decompose a facial expression into one or more of 44 AUs, that produced the expression in question [12]. Recent work on facial AU detection applying biologically inspired algorithms has used the technologies as Artificial Neural Network (ANN), Support Vector Machine (SVM), Bayesian Networks, k-Nearest Neighbor. Also three more methods of facial emotion recognition are described in [16] as: Holistic Matching Methods, Feature-based (structural) Methods and Hybrid Methods. These methods are explained in [16] along with the applications.

III. PROPOSED METHODOLOGY

The actual method is based on studying the underlying AUs that are involved in expressing the different types of emotions. We identify the specific AUs whose movements we wish to observe using small luminous markers that are placed on the face of the subject. So, before applying the PSO algorithm to human's face in one of the captured video clip there is necessary to plot action units on human's face. Then by simply observing and analyzing the changes in the positions of the action units on the face we can identify the emotion which is expressed in a video clip.

A. Position Of Action units

Figure 3. shows position of 14 action units which are generally located on the face of human being. These action units are considered to be the smallest visually perceptible facial movements. The location of action units mainly involves eyebrows, eyes and lips part of human's face. As action units are independent of any interpretation, they can be used as the basis for recognition of basic emotions. It is an unmistakable means of describing all possible movements of face in 46 action points.

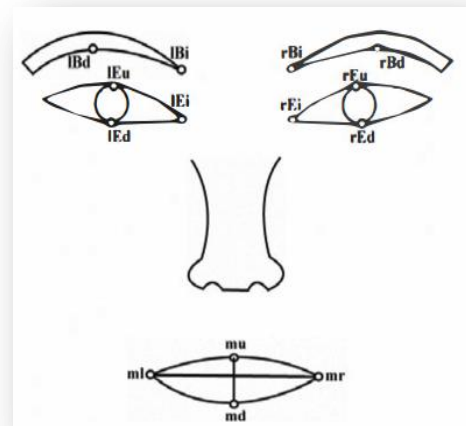


Figure 3: Position of Action Units

B. The original PSO Algorithm

PSO in principle is a parallel multi-agent search technique. Particles are conceptual entities which fly through the multi-dimensional search space. At any particular instant, each particle has a position and a velocity. The position vector of a particle with respect to the origin of the search space represents a trial solution of the search problem [14].

The population of such particles is called a 'swarm' S . A neighborhood relation N is defined in the swarm. N determines for any two particles P_i and P_j whether they are neighbors or not. Thus for any particle P , a neighborhood can be assigned as $N(P)$, containing all the neighbors of that particle. Different neighborhood topologies and their effect on the swarm performance have been discussed. The PSO used in this work, implicitly uses a so-called fully connected neighborhood topology (or *gbest*). Every particle is a neighbor of every other particle [14]. Each particle P has two state variables:

1. Its current position $x(t)$.
2. Its current velocity $v(t)$.

Particle Swarm Optimization (PSO) is a population-based search algorithm initially designed to simulate the social behavior of birds in a flock as they fly in search of food. A PSO algorithm maintains a swarm of particles, where each particle represents a potential solution [14]. In a PSO system, a swarm of individuals (called particles) fly through the search space. Each particle represents a candidate solution to the optimization problem. The position of a particle is influenced by the best position visited by itself (i.e. its own experience) and the position of the best particle in its neighborhood (i.e. the experience of neighboring particles). When the neighborhood of a particle is the entire 24 swarm, the best position in the neighborhood is referred to as the global best particle, and the resulting algorithm is referred to as a *gbest* PSO. When smaller neighborhoods are used, the algorithm is generally referred to as a *lbest* PSO. The performance of each particle is measured using a fitness function that varies depending on the optimization problem [15].

Each Particles are "flown" through a multi-dimensional search space, where the position of a particle is adjusted according to two factors:

1. Its own successful experience
2. The successful experiences of its neighbors.

In [15] they proposed following PSO and its modification as GPSO.

Let $x_i(t)$ denote the position of particle i at time t . The position of the particle is changed by adding a velocity, $v_i(t+1)$ to the current position.

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (1)$$

where $x_i(0)$ is generated randomly from the range $[x_{\min}, x_{\max}]$.

It is the velocity vector that drives the optimization process, and reflects both the experience of the particle and the experiences of its neighbors. The experiential knowledge of the Particle is referred to as the cognitive

component, and is proportional to the distance of the particle from its own best position [14]. The socially exchanged information is referred to as the social component of the velocity equation. Originally, two PSO algorithms were developed, which differ in the size of their neighborhoods. These two algorithms are known as *gbest* and *lbest* [14].

For the global best PSO, the neighborhood for each particle is the entire swarm. The social networking employed by *gbest* PSO reflects the star topology, where the social component of the velocity equation reflects the information obtained from the entire swarm [14]. In this case, the social component is the best position found by the swarm, represented as $\hat{y}(t)$. For *gbest* PSO, the velocity of particle i is calculated as in equation (2).

$$v_i(t+1) = v_i(t) + c_1 r_1(t)[y_i(t) - x_i(t)] + c_2 r_2(t)[\hat{y}(t) - x_i(t)] \quad (2)$$

where, $v_i(t)$ is the velocity of particle i in a given dimension at time t . $x_i(t)$ is the position of particle i in a given dimension at time t . c_1 and c_2 are positive acceleration constants. $r_1(t)$, $r_2(t)$ are random values in the range $[0, 1]$ generated at time t and $y_i(t)$ is the best position so far found by particle i .

For minimization problem, the personal best at the next time step, $t+1$, is calculated as:

$$y_i(t+1) = \begin{cases} y_i(t) & \text{if } f(x_i(t+1)) \geq f(y_i(t)) \\ x_i(t+1) & \text{if } f(x_i(t+1)) < f(y_i(t)) \end{cases} \quad (3)$$

Where $f: R^n \rightarrow R$ is the fitness (or objective) function, which measures how close the corresponding solution is to the optimum. Following algorithm Summarizes the *gbest* PSO algorithm. The local best PSO, *lbest*, is similar to the *gbest* except that it uses a ring social network topology, where smaller neighborhoods are defined for each particle [14]. The social component reflects the information exchanged within the neighborhood of the particle. Thus, the velocity update equation is modified as in equation (4).

$$v_i(t+1) = v_i(t) + c_1 r_1(t)[y_i(t) - x_i(t)] + c_2 r_2(t)[\hat{y}_i(t) - x_i(t)] \quad (4)$$

Where $\hat{y}_i(t)$ is the best position found by the neighborhood of particle i in a given dimension.

• PSO (Global best) algorithm.

1. Create and initialize an n – dimensional swarm;
2. repeat
3. for each particle $i = 1, \dots, n$ do
4. //set the personal best position
5. if $f(x_i) < f(y_i)$ then
6. $y_i = x_i$;
7. end
8. // set the global best position
9. if $f(y_i) < f(\hat{y})$ then
10. $\hat{y} = y_i$;

11. end
12. end
13. for each particle $i = 1, \dots, n$ do
14. update the velocity using equation (2);
15. update the position using equation (1);
16. end
17. until stopping condition is true

The two versions of PSO algorithms are similar in the sense that the social component of the velocity updates causes both to move towards the global best. There are two main differences:

- Due to the larger particle interconnectivity of *gbest*, it converges faster than *lbest*. This convergence comes at the cost of less diversity.
- Due to the larger diversity in *lbest*, which results in more coverage of the search space, it is less prone to being trapped in local minima [15].

So, in [15] a modified PSO algorithm, which is called as Guided Particle Swarm Optimization (GPSO) algorithm is further defined which is applied to the detection of facial emotion of a person in a captured video clip.

C. The GPSO Algorithm

The emotion detection problem is a search problem, where at each point, there is need to identify which of the possible emotions does the current human's facial expression represents. Thus, clearly emotion detection lends itself as a possible candidate for PSO application. However, in order to apply PSO to solve the emotion detection problem, we need to first define the various parameters of the algorithm in relation to the problem.

We have stated our approach to the emotion detection problem, which is basically to monitor the changes in the positions of the action units, placed on the face of a subject over a period of time, from which we can then determine the emotion expressed at each point in time. With this in mind, we define the parameters of the PSO as follows:

Definition 1: Search space and its dimension:

Let the Action Units (AUs), be denoted by, q_1, q_2, \dots, q_n .

Let D_1, D_2, \dots, D_n represent the domains of the AUs, q_1, q_2, \dots, q_n , respectively. That is D_j represents a 2 dimensional rectangular neighborhood window consisting of the possible points that q_j can be assigned to. Then the search space is a n -tuple, R^n , given by:

$$R^n = (D_1, D_2, \dots, D_n) \quad (5)$$

The dimension of the search space is n , where n is the number of action units being observed.

Definition 2: Particle, its position and velocity:

A particle P is an abstract object in the R^n search space that has a position and a velocity and represents a possible solution.

The position, $x_i(t)$ of a particle, P_i at time t , is a complete assignment of values $(val_1, val_2, \dots, val_n)$, where $val_j \in D_j$. Thus, $x_i(t)$ is a vector, $(val_1, val_2, \dots, val_n)$.

The velocity, $v_i(t)$ of particle i at time t is an n -tuple (v_1, v_2, \dots, v_n) where v_j represents the velocity of the particle in dimension D_j .

There are two peculiar issues that make the emotion detection problem a little different from normal problems to which PSO is applied. First, in normal PSO problems, there is usually one target that all particles in the swarm are trying to reach. In our particular case however, there are a number of possible emotions and any one of them could be encountered at any time. In order to solve this multi-target problem, we propose to have multiple swarms, one for each possible emotion. Since each swarm has a different target to reach, the objective function of each swarm must be defined differently. We define the objective function of each swarm as the Euclidean distance between its current position and its target. For example, the following is the definition of the objective function for the swarm that is targeting the happy emotion.

Definition 3: Objective function for the happy-targeting swarm:

Let $S = (s_1, s_2, \dots, s_n)$ represent the happy emotion. Then the objective function for the happy swarm, $f: R^n \rightarrow R$ is defined as:

$$f_s(X_i(t)) = |X_i(t) - S|$$

$$= \sqrt{(x_1 - s_1)^2 + (x_2 - s_2)^2 + \dots + (x_n - s_n)^2} \quad (6)$$

• The GPSO algorithm.

1. Create and initialize m swarms of n dimensions.
2. // m is the # of different emotions
3. // n is the # of action units being tracked.
4. Add a particle, Q , representing the positions of the AUs in each swarm
5. Read the approximate position of each emotion from a file.
6. For each frame of the video, do
7. For each swarm do
8. for each particle in the swarm do
9. //set the personal best position, y_i
10. if $f(x_i) < f(y_i)$ then
11. $y_i = x_i$;
12. end
13. //set global best to be position of Q .
14. end
15. for each particle in the swarm do
16. update the velocity using (7);
17. update the position using (1);
18. end
19. compute the distance of the swarm from its target emotion
20. end

21. declare the target emotion of the swarm whose distance from its target is the shortest as the emotion being expressed in the current frame.
22. end

In each iteration of the PSO algorithm, each swarm will update the positions of its particles as usual. These positions are then compared to find the swarm that is closest to its target. Such a swarm is considered to have found a solution.

The second issue that makes the emotion-detection problem a little different from normal PSO problems is that in this case we have the data about the positions of the action units. If the particles can take advantage of this knowledge, then they are likely to reach their target sooner than if they rely solely on their cognitive and experiential knowledge. Accordingly, in [15] they proposed the following changes to the algorithm:

The positions of the AUs should always be represented as one of the particles in each swarm. That is, let Q be a particle whose position $X_q(t) = (q_1, q_2, \dots, q_n)$, where q_1, q_2, \dots, q_n are the positions of the n AUs respectively. Then Q must be included as a particle in each swarm.

The velocity update equation is changed from (2) to (7), where the position of Q is always regarded as the global best.

$$v_i(t+1) = v_i(t) + c_1 r_1(t)[y_i(t) - x_i(t)] + c_2 r_2(t)[q(t) - x_i(t)] \quad (7)$$

With these proposed changes, the particles are effectively guided to converge towards the path of the action units. Accordingly, this modified version of the algorithm is called as the Guided Particle Swarm Optimization (GPSO) algorithm [15]. The GPSO algorithm stated above is implemented using C# programming language under the .NET development framework. In the actual detection system it took a video clip as input, the digitized data for the video clip and the positions of the AUs corresponding to the various emotions as captured in the training session. The system initializes a swarm by creating random particles within the domain of each of the Action units. The GPSO algorithm was then executed to detect the emotions expressed in each frame of the video clip. The detected emotion is visually displayed on the screen. For this study, they considered all the six universal basic emotions, namely happy, sad, surprise, disgust, anger and fear. These six, plus the neutral state gives seven possible states that the GPSO system can detect [15].

IV. ADVANTAGES AND DISADVANTAGES

A. Advantages

- 1) The GPSO and PSO is based on the intelligence. So, it can be applied into both scientific research and engineering use.
- 2) GPSO have no overlapping and mutation calculation. The search can be carried out by the speed of the

particle. During the development of several generations, only the most optimist particle can transmit information onto the other particles, and the speed of the researching is very fast.

- 3) The calculation in PSO is very simple. Compared with the other developing calculations, it occupies the bigger optimization ability and it can be completed easily.
- 4) GPSOPSO adopts the real number code, and it is decided directly by the solution. The number of the dimension is equal to the constant of the solution.

B. Disadvantages

- 1) The PSO method easily suffers from the partial optimism, which causes the less exact at the regulation of its speed and the direction.
- 2) The method of particle swarm optimization can not work out the problems of scattering and optimization.
- 3) The method can not work out the problems of non-coordinate system, such as the solution to the energy field and the moving rules of the particles in the energy field.

V. CONCLUSION

The research in facial emotion recognition is an exciting area for many years to come and will keep many scientists and engineers busy. This paper provides the readers a better understanding about study of PSO and GPSO algorithm which can be the best method for the emotion detection. In this paper we presented a modification of the Particle Swarm Optimization (PSO) algorithm which is called as Guided Particle Swarm Optimization (GPSO) algorithm for the purpose of emotion detection. In this paper we have studied the PSO and GPSO algorithm which can be further applied in future to actual identification of emotion using facial expressions. In future work, we will analyze the system by actually applying the algorithm to a video clip or to picture and also the system shall improve to work with video clips in which subjects do not have AUs marked on their faces.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude towards guide Dr. G. S. Sable, professor at SPWEC, Aurangabad for his time to time help and guidance.

REFERENCES

- [1] D. Keltner and P. Ekman, "Facial expression of emotion", Handbook of emotions, M. Lewis and J.M. Haviland-Jones, Eds. Guilford Press, New York, 2000, pp. 236-249.
- [2] J. Kennedy, R. Eberhart, "Particle swarm optimization", Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942-1948.
- [3] K. P. Wang, L. Huang, C. G. Zhou, and W. Pang, "Particle Swarm Optimization for Travelling Salesman Problem", Proceedings of International Conference on Machine Learning and Cybernetics, 2003, pp. 1583-1585.
- [4] V. Miranda and N. Fonseca, "EPSO -- Evolutionary Particle Swarm Optimization, a New Algorithm with Applications in Power Systems", Proceedings of the Asia Pacific IEEE/PES Transmission and Distribution Conference and Exhibition, 2002, vol. 2, pp. 745-750.
- [5] M. Settles and B. Rylander, "Neural Network Learning using Particle Swarm Optimizers", Advances in Information Science and Soft Computing, 2002, pp. 224-226.
- [6] M.G. Omran, A.P. Engelbrecht, and A. Salman, "Particle Swarm Optimization Method for Image Clustering", International Journal on Pattern Recognition and Artificial Intelligence, 2005, vol. 19, no.3, pp. 297-322.
- [7] D.W. van der Merwe and A.P. Engelbrecht, "Data Clustering using Particle Swarm Optimization", Proc. of IEEE Congress on Evolutionary Computation, 2003, vol. 1, pp. 215-220.
- [8] X. Xiao, E.R. Dow, R.C. Eberhart, Z. Ben Miled, and R.J. Oppelt, "Gene Clustering using Self-Organizing Maps and Particle Swarm Optimization", Online Proceedings of the Second IEEE International Workshop on High Performance Computational Biology, 2003.
- [9] B.B. Thompson, R.J. Marks, M.A. El-Sharkawi, W.J. Fox, and R.T. Miyamoto, "Inversion of Neural Network Underwater Acoustic Model for Estimation of Bottom Parameters using Modified Particle Swarm Optimizer", in Proc. of the International Joint Conference on Neural Networks, 2003, pp. 1301-1306.
- [10] Salman, I. Ahmad, and S. Al-Madani, "Particle Swarm Optimization for Task Assignment Problem", Microprocessors and Microsystems, 2002, vol. 26, no. 8, pp. 363-371.
- [11] C.A. Coello Coello, E.H.N. Luna, and A.H.N. Aguirre, "Use of Particle Swarm Optimization to Design Combinational Logic Circuits", Lecture Notes in Computer Science, Springer-Verlag, 2003, vol. 2606, pp. 398-409.
- [12] M. F. Valstar and M. Pantic, "Biologically vs. Logic Inspired Encoding of Facial Actions and Emotions in Video". IEEE Inter. Conference on Multimedia and Expo, 2006, pp. 325--328.
- [13] M. Pantic and L.J.M. Rothkrantz, "Toward an Affect-Sensitive Multimodal Human-computer Interaction", Proceedings of the IEEE, 2003, vol. 91, no. 9, pp. 1370-1390.
- [14] J.Kennedy, R. Mendes, "Population Structure and Particle Performance", in Proceedings of the IEEE congress on Evolutionary Computation, 2002, pp. 1671-1676.
- [15] Bashir Mohammed Ghandi, R. Nagarajan, Hazry Desa, "Facial emotion detection using guided particle swarm optimization (GPSO)", the International Conference on Man-Machine Systems (ICoMMS), 11 – 13 October 2009, pp. 2A1-1 - 2A1-5
- [16] Divyarajsinh N. Parmar, Brijesh B. Mehta, "Face Recognition Methods & Applications", Divyarajsinh N Parmar et al, Int.J.Computer Technology & Applications, Vol 4 (1), Jan-Feb 2013, pp 84-86.
- [17] A. Habibizad navin, Mir Kamal Mirnia, "A New Algorithm to Classify Face Emotions through Eye and Lip Features by using Particle Swarm Optimization", 4th International Conference on Computer Modeling and Simulation, vol.22, 2012, pp. 268-274