

Learning Individualized Facial Expressions in an Avatar with PSO and Tabu Search

Evan Husk, Avelino J. Gonzalez and Sumanta Pattanaik

Dept. of EE and CS – CS Division
University of Central Florida
Orlando, FL

Evan.Husk@knights.ucf.edu, Avelino.Gonzalez@ucf.edu, sumanta@cs.ucf.edu

Abstract

This paper describes a method for automatically imitating a particular facial expression in an avatar through a hybrid Particle Swarm Optimization – Tabu Search algorithm. The muscular structures of the facial expressions are measured by Ekman and Friesen's Facial Action Coding System (FACS). Using a neutral expression as a reference, the minute movements of the Action Units, used in FACS, are automatically tracked and mapped onto the avatar using a hybrid method. The hybrid algorithm is composed of Particle Swarm Optimization algorithm and Tabu Search. Distinguishable features portrayed on the avatar ensure a personalized, realistic imitation of the facial expressions. To evaluate the feasibility of using PSO-TS in this approach, a fundamental proof-of-concept test is employed on the system using the OGRE avatar. Results are described and discussed.

1. Introduction and Background

In the future, having an avatar of oneself that looks and sounds like its human “master” could be a highly useful commodity. If one cannot (or does not wish to) attend a meeting, he/she could send his/her avatar to attend in his/her place and even possibly (although farther in the future), speak and make commitments for its master. Of course, its intelligence would be the primary element of the avatar's being; however, that is not our objective here. Almost as important, if it were to credibly stand in for its human master, it must directly resemble the person it is representing. Advances in making avatars appear real have been made [Hung et al. 2010]. See Figure 1 for an example of an avatar built to imitate a real person.

Zalewski and Gong [2005] note that no two emotional expressions in humans are alike. That is, no two people

smile in the exact same manner because there are miniscule details that make each unique. When building graphic representations of specific humans, it is important to reflect such personal characteristics as realistically as possible. However, the facial expressions of a particular individual, especially when expressing an emotion, are still implemented mostly in a generic, “cookie cutter” manner.



Figure 1: A Lifelike avatar for a Specific Individual

Ekman [1993] gives an overview on the classifications of emotions and describes many psychological aspects of emotional expressions. He established a link between humans and facial expressions, explaining how each individual expresses his/her current emotional state. He found that although cultures differ in many aspects, basic emotional expressions are, for the most part, all universally understood and expressed in the same manner.

The works described by Zalewski and Gong [2005] and by Mori et al. [2011] have been geared towards recognizing slight movements that make certain expressions personal. Lin et al. [2008], Mipiperis et al. [2008], and Yang and Banu [2011] used systems that incorporated avatars; however, these were used to help detect and classify facial expressions, not imitate and recreate them. Other contributions to the state-of-the-art

have been made by Cosker et al. [2011] and Velusami et al. [2011].

Mori et al. [2011] built a system that incorporates an avatar that expresses emotions in a natural environment (i.e. a daily conversation). The authors describe how emotional facial expressions can be expressed in an avatar using utterances as an input. That is, the avatar attempts to take on the emotion expressed in a given sound bite. Their system adds a personalized aspect in the recreation of the emotion by considering unique features of the face. To accomplish this personalization of expressions, the authors take into account the subtle differences in expressions. However, significant manual effort is involved in this manipulation - exactly what we are trying to avoid here.

There is a justified need for personalized expressions in avatars for numerous applications. This paper presents a means to apply machine learning techniques to learn how to express emotions for the avatar simply by learning these expressions from a photo or video of the person of interest on a pixel-by-pixel basis. Our objective is to use machine learning techniques to learn the specific facial expression on the human master, as recorded on a photograph or video (photographs were used here), and have the system set the “slider” values of the *working* avatar copy. This is to be done by comparing the images on a pixel-by-pixel basis.

Nevertheless, for reasons of practicality, a lifelike avatar was not used in this research. Instead, the concept was tested using the OGRE avatar [OGRE], which is a cartoon-like figure. Nevertheless, its open access and use of sliders to move the facial muscles provided an excellent intermediate step to test our algorithms. In our work, two faces of the OGRE avatar were used, the *target* copy and the *working* copy. The target OGRE face is the desired look of the avatar. Sliders were used to pre-select the facial muscles and achieve a unique specific expression. Thus, the target avatar face can be considered to be the equivalent of the photograph of a particular human depicting the desired expression. The working copy is the one that is to learn how to achieve the look of the target copy. In our lifelike avatar analogy, the working copy would be the avatar itself that is to be made to resemble the human’s expression in the photo (the target copy). See Figure 2 for a depiction of the OGRE avatar face.

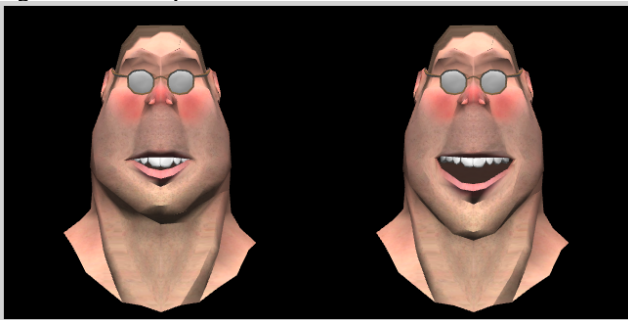


Figure 2: OGRE Tool [OGRE]: Working Avatar face (left) and Target Avatar face (right)

1.1 FACS and Action Units

In order to build a learning algorithm that can do the above, the muscular structure and features of the avatar’s face have to be defined in a way that can be measured. This was done using a form of Ekman and Friesen’s [1978] *Facial Action Coding System* (FACS). By designating Action Units (AUs) involved with the face, the movements and muscular contractions of the face can be detected and tracked to help evaluate the slight differences that make an avatar’s facial expressions unique.

Ekman and Friesen [1978] developed FACS to allow for a quantitative representation of facial expressions. At the time, the system was needed to further study emotional expressions and social interaction between humans [Donato et al. 1999]. Ekman and Friesen noticed that certain muscular contractions were the basis of rearranging the face in a way to express an emotional state. The facial structure was decomposed into segments to monitor movements in specific portions of the face. Thus, they decided to base their system on these muscular contractions by pinpointing and tracking key locations on the face that encountered significant changes in the process of expressing an emotion [Donato et al. 1999].

The specific locations that accurately express the independent motion of features in the face are more formally referred to as the Action Units (AUs). The FACS system incorporates 44 AUs to accurately track and measure the facial differences encountered when expressing an emotion [Cosker et al. 2011]. FACS is based on six basic emotions: anger, disgust, fear, happiness, sadness, and surprise.

1.2 Mapping Action Units

In order to correctly map the AUs defined in FACS, an algorithm is typically used that may incorporate machine learning in the process. The AUs on the OGRE avatar are controlled by the sliders. Therefore, when the algorithm manipulates the sliders, the corresponding AUs are also manipulated. This algorithm allows the mapping process to take place efficiently and effectively. An algorithm that has been found to be successful in search and optimization problems is Particle Swarm Optimization (PSO) [Kennedy and Eberhart, 1995]. Variations of PSO have been used in many applications that require an optimal solution [Ghandi et al. 2009]. In their work, Ghandi et al. make modifications to incorporate PSO in emotion detection problems. Another search algorithm used in optimization problems is Glover’s Tabu Search (TS) [1989, 1990]. Unlike PSO, Tabu Search is a local search algorithm that takes advantage of a memory structure to log solutions that were previously visited [Glover, 1989; 1990; Bekrar et al. 2011; Zhang and Wu, 2011]. Because of its local search characteristics and use of memory, TS has been “hybridized” with other algorithms to increase its robustness in certain applications [Thangaraj et al. 2011;

Bekrar et al. 2011; Zhang and Wu, 2011; 2012; Li et al. 2010; Mhamdi et al. 2011]. Because of space limitations, PSO and TS are not described here but the reader is referred to their cited original descriptions.

1.3 Objectives of Our Work

Although PSO can successfully search throughout a large search space and find an optimal solution, it often prematurely converges toward local optima. This means the algorithm often is subject to partial optimization. Instead of finding the global optimal solution, it finds the local optimal solution within the swarm [Bai, 2010]. A specific example of this premature convergence is seen in [Puklavage et al. 2010]. Their mechanism uses PSO to map the AUs on the OGRE avatar using the PSO mechanism, which successfully converges to the optimal solution for the happy, sad, fear, and surprise test faces [Puklavage et al. 2010]. However, the weakness of PSO was exposed when their system failed to accurately converge to the target angry face of the OGRE avatar. In the specific case of converging to the angry face, PSO continually fails to find the optimal solution within the search space because it becomes trapped in a local minimum.

This prior work by Puklavage et al. [2010] attempted to use the same OGRE avatar (albeit an earlier version) and PSO to achieve the same objectives described above – that of using machine learning to automatically learn to replicate personal facial expressions from photographs of the avatar’s human counterpart. Our research, therefore, is a follow up on Puklavage et al’s work in an attempt to improve upon their results.

Our approach was to make a hybrid PSO-TS learning algorithm, where the TS part compensated for the limitations found to exist in the PSO approach to this problem. Clearly, the hybrid combination of PSO and TS is not novel, as several others have implemented it as well as variations thereof. Nevertheless, we describe our application of this technique to solve a problem that was not completely solved using PSO by itself.

2 The PSO-TS Hybrid System

Our hybrid PSO-TS algorithm is described in Figure 3. This hybrid PSO-TS algorithm combines a slightly modified form of PSO with a standard Tabu Search. The flowchart of Figure 3 provides a top-level overview of the architecture of the algorithm. As depicted in the flowchart, PSO serves as the driving force of the PSO-TS algorithm. Unlike other models where the search population is randomly halved and TS and PSO are independently operated [Zhang and Wu, 2011; 2012], our hybrid PSO-TS embeds TS within PSO. The diversification of the search provided by TS helps PSO avoid its limitation of premature convergence toward local optima. The PSO algorithm drives the overall search and begins the process

by a random search through the solution space. Once initial *pbests* and a global *gbest* are obtained, the *pbests* are passed to Tabu Search to explore the nearby area of the swarm. TS takes in these *pbests* and establishes a local search boundary centered on each *pbest*. The search examines the nearby area for a potential better solution. If a better solution is found, TS returns this updated solution to PSO. PSO then updates the swarm based on the best solution found thus far. The *pbest* and *gbest* along with the particles’ velocities and positions are updated. The process continues in this manner until the stopping criteria is met. Overall, the combination of PSO and TS joins the strength of PSO’s global search with TS’s local search to help diversify the search and overcome PSO’s premature convergence to poor quality local optima [Zhang and Wu, 2011; Bekrar et al. 2011].

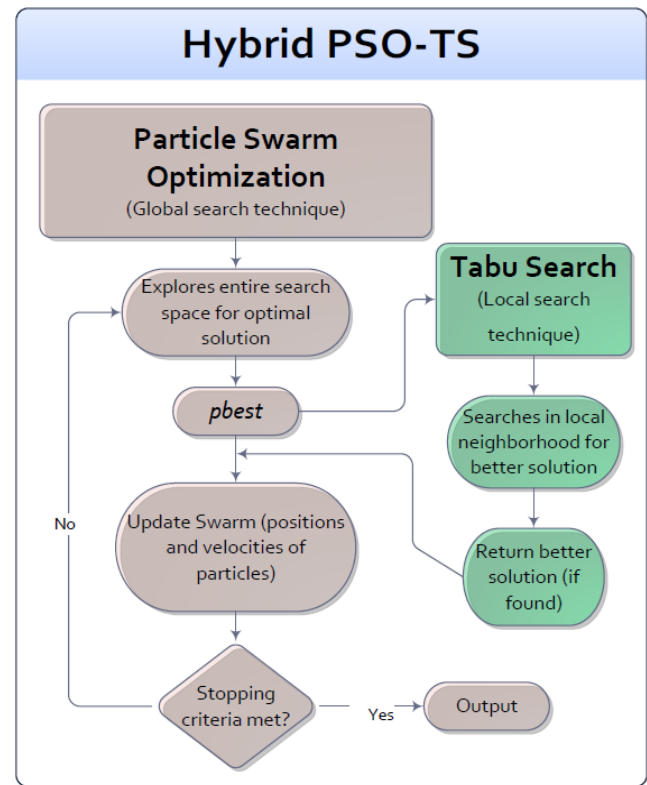


Figure 3: Top Level Flowchart of PSO-TS

3 The OGRE Application

We first describe our use of OGRE as the testbed to test our algorithm. To do this, we developed the OGRE Algorithm Comparison Tool (OACT). To begin the development of OACT, the OGRE Official Demos Distribution v1.7.0 (Windows) was downloaded [OGRE]. The OGRE face is centered in the middle of the screen with 18 sliders available in the upper left-hand corner. Once the “Manual” option is selected, the OGRE face stops its automated movements and provides the user with

full control of the facial musculature of the OGRE avatar. The sliders are individually labeled to describe their corresponding action.

The overall architecture of the downloaded Facial Animation demo was transferred via JavaScript to a WebGL application to allow for the same manipulation of the OGRE face via a set of 18 sliders. Our OACT provides a similar layout to the Facial Animation demo. As in our tool, Puklavage et al. [2010] also based their PSO mechanism on the OGRE Facial Animation demo. However, unlike their work, our OACT is not directly embedded in, or dependent on, the OGRE demo.

When developing OACT, the facial models and textures used in the Facial Animation demo were directly implanted in the WebGL application to ensure the same OGRE avatar face was used. Unlike the OGRE Facial Animation demo and the mechanism used by Puklavage et al., our tool uses two OGRE avatar faces. These faces are set in a horizontal format to allow the application to provide real-time feedback through the expressions on the OGRE avatars. The layout of our tool is provided in Figure 4 below. The face on the left is the *working* face while the face on the right is the *target* face. The musculature of the target face can be manipulated via the sliders on the right side of the tool. There are 19 sliders, but only 18 of them are used to manipulate the avatar face. The neutral slider under the “Expressions” heading is incorporated strictly to resemble the updated version of the OGRE demo, to allow for future improvements of the tool. The 18 sliders can take in a value between 0 and 1, corresponding to the intensity of the corresponding expression.

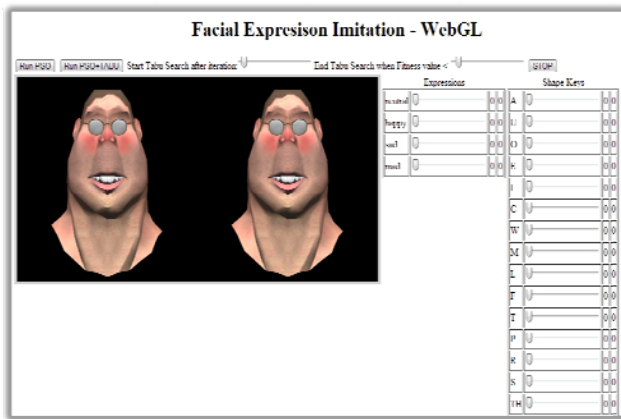


Figure 4: OGRE Algorithm Comparison Tool

Deviating from the standard layout of the OGRE demo, some instruments were added to assist in testing. Placed at the top of the OACT are three buttons and two sliders. The “Run PSO” and “Run PSO+TABU” buttons run their respective algorithms. It should be noted that the modular design of the hybrid PSO-TS algorithm allows the algorithms to run independently. Thus, the “Run PSO” button will only run PSO and the “Run PSO-TABU”

button will run the hybrid PSO-TS algorithm. The “STOP” button immediately halts the process and displays the best facial match up to that point in the search. The “Start Tabu Search after iteration:” slider is used to activate Tabu Search after a specified number of iterations. The “End Tabu Search when Fitness <” slider is used to terminate the local Tabu Search once a fitness threshold, discussed in the following section, is reached. Both of these sliders are



Figure 5: OACT Convergence Feedback

only applicable to the PSO-TS search method. Again, these additional buttons and sliders are used for testing purposes only and do not directly control the avatar’s facial structure. An important factor considered while developing OACT was the search status and convergence feedback. Once either search is activated, real time feedback of the progress of the search is provided through the working avatar face. Keeping in mind that the working face is on the left and the target face is on the right, the avatar faces in Figure 5 demonstrate the progression of the search as it converges toward the optimal solution. The top set of OGRE faces shows a working avatar face that is greatly deformed. At this point, the search has just begun and the algorithm is just beginning to explore the search space. As the search finds better solutions, it continually moves toward the target face as seen in the middle faces. Finally, once the optimal solution has been found, the working and target avatar faces closely match, as depicted in the bottom pair of faces.

4 Experiments and Results

The testing involved two phases. We describe each separately in the following subsections.

4.1 Phase I: Establishing a Baseline

The purpose of Phase I was to establish a baseline for our PSO-only system that matched the results obtained by Puklavage et al. [2010]. This was to preclude the possibility that our specific implementation of PSO, our use of the OACT, and/or the new version of the OGRE avatar face may have introduced experimental bias not reflected in the work of Puklavage et al. It should be noted that the new version of OGRE used in our implementation has updated glasses that “shade” the eyes. However, this does not affect the comparison of our results because none of the facial expressions tested incorporate the eyes of the

avatar. Our version of PSO, called “OACT PSO” for lack of a better name, was run on the five faces imitated by Puklavage et al. The results are shown in Figure 6.

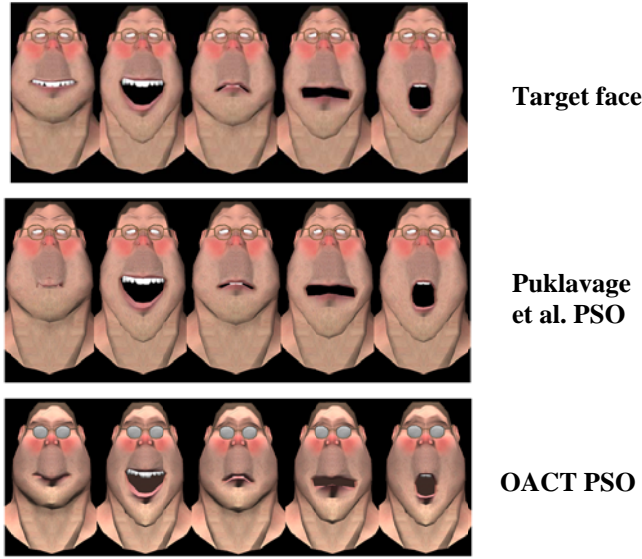


Figure 6: Phase I Visual Results

It can be seen that, as in Puklavage et al’s work, the rightmost four faces converged successfully. This was done in the same number of iterations (250), and using the same swarm size (50). More importantly, it can be easily seen that the leftmost face, that depicting anger, failed to converge, as in Puklavage et al’s case, albeit resulting in a different expression altogether. This confirms that the OACT PSO meets the baseline requirements. Nevertheless, one should note that the four faces that successfully converged with OACT PSO are slightly different than the target faces because of the different versions of OGRE.

4.2 Phase II: PSO-TS Performance

This Phase of the testing evaluated the core objective of our research. Is the combination of PSO with TS able to overcome the deficiencies experienced by Puklavage et al? Using OACT and applying PSO-TS, the same five OGRE faces were subjected to the process of reproducing the five target faces in a pixel-by-pixel manner. Figure 7 graphically depicts the results. It can be seen that the angry face (the left-most face in Fig. 7) generated by our PSO-TS algorithm clearly resembles the target angry face much more closely than did Puklavage et al’s PSO.

The graphs in Figure 8 reflect data collected on the fitness of the OGRE *working* faces computed during the execution of both algorithms. The fitness value (0-3) directly reflects the pixel-by-pixel differences between the *working* and *target* avatar faces by calculating the Manhattan distance between two reference points. Higher fitness values reflect greater differences in the two faces. It should be noted that the initial fitness is dependent on the random initialization of the swarm. Therefore, the initial fitness values

significantly vary based on the random initialization of PSO. These figures clearly show that the PSO-TS algorithm converged much more quickly and with significantly less error than did Puklavage et al’s PSO implementation.

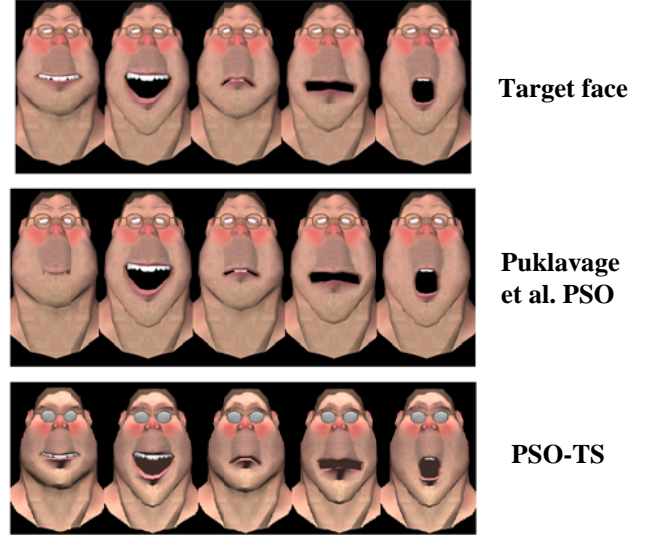
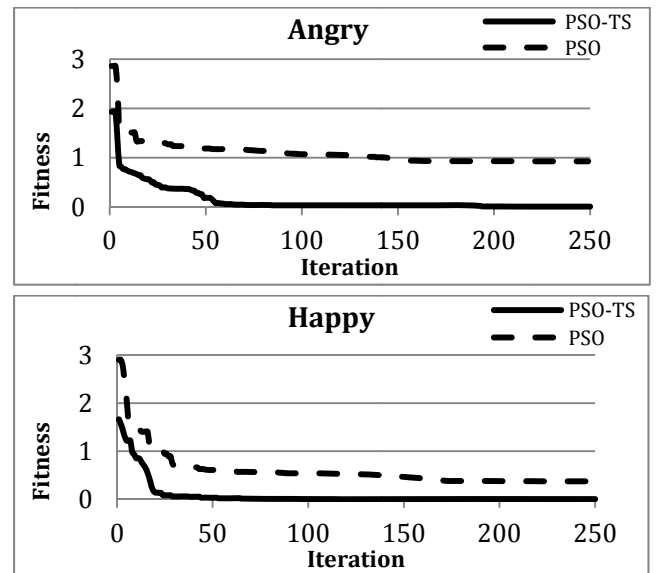


Figure 7: Phase II Visual Results

5. Conclusion

In conclusion, the data shows that the PSO-TS algorithm represents a significant improvement over the earlier work by Puklavage et al. [2010]. While the PSO-TS algorithm resulted in some differences in the faces, these were considered negligible. However, this assumption needs to be tested in the future with human subjects for confirmation. The use of the OGRE avatar represents an intermediate step towards our ultimate goal of being able to reproduce in an avatar the personal facial expressions of specific individuals.



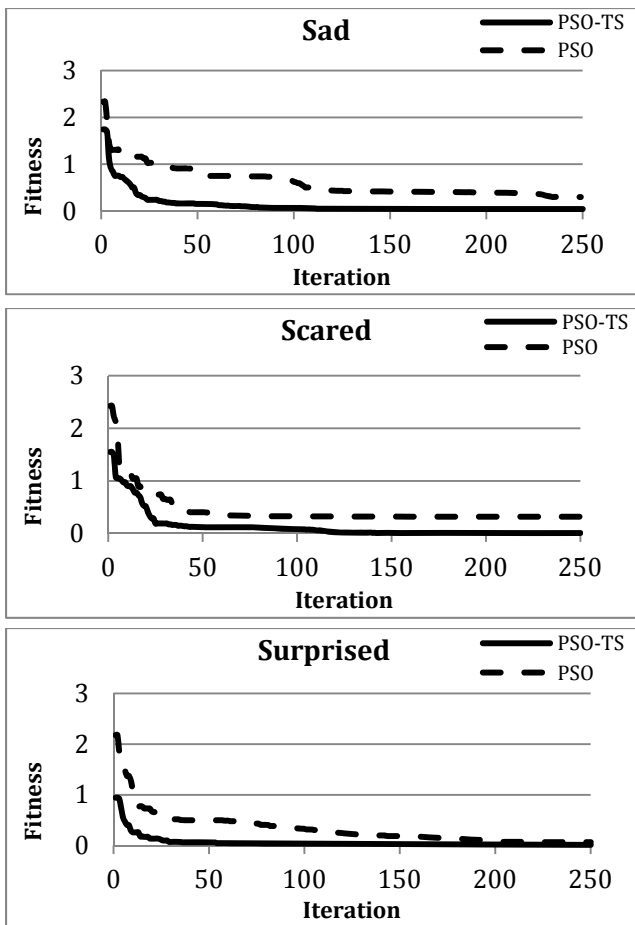


Figure 8: Convergence Comparison of PSO-TS to Puklavage et al's PSO

6. References

- Bai, Q., "Analysis of Particle Swarm Optimization Algorithm," *Computer and Information Science*, 3(1), pp. 180-184, 2010.
- Bekrar, A., Chaabane, S., Trentesaux, D., Bornschlegell, A., Pelle, J. and Harmand, S., "Hybrid PSO-tabu search for constrained non-linear optimization problems," *International Conference on Swarm Intelligence*, Cergy, France, 2011.
- Cosker, D., Krumhuber, E. and Hilton, A., "FACS valid 3D dynamic action unit database with applications to 3D dynamic morphable facial modeling," *IEEE International Conference on Computer Vision* pp. 2296-2303, 2011.
- Donato, G., Bartlett, M.S., Hager, J.C., Ekman, P. and Sejnowski T.J., "Classifying facial actions," *IEEE Trans. on Pattern Anal. & Machine Intelligence*, 21(10), pp. 974-989, 1999
- Ekman P. and Friesen, W. "Facial action coding system: A technique for the measurement of facial movement". Palo Alto, CA: Consulting Psychologists Press, 1978.
- Ekman, P., "Facial expression and emotion," *The American Psychological Association, Inc.*, vol. 48, no. 4, pp. 384-392, 1993.
- Ghandi, B.M., Nagarajan, R. and Desa, H., "Particle swarm optimization algorithm for facial emotion detection," *IEEE Symp. on Industrial Electronics & Appl.*, vol. 2, pp. 595-599, 2009.
- Glover, F., "Tabu search: Part I," *ORSA Journal on Computing*, 1(3), pp. 190-206, 1989.
- Glover, F., "Tabu search: Part II," *ORSA Journal on Computing*, 2(1), pp. 4-32, 1990.
- Hung, V., Gonzalez, A. J. and DeMara, R. F., "Dialog Management for Rapid-Prototyping of Speech-Based Training Agents", *Proceedings of the Interservice/Industry Training Simulation and Education Conference*, Orlando, FL Dec. 2, 2010
- Kennedy J. and Eberhart, R. "Particle swarm optimization," *Proceedings of IEEE International Conference on Neural Networks*, Perth, Australia, pp. 1942-1948, 1995.
- Li, J., Pan, Q., Xie, S., Jia, B. and Wang, Y., "A hybrid particle swarm optimization and tabu search algorithm for flexible job-shop scheduling problem," *International Journal of Computer Theory and Engineering*, 2(2), pp. 189-194, 2010.
- Lin, C.J., Chu, C.H., Lee, C.Y. and Huang, Y.T. "2D/3D face recognition using neural networks based on hybrid taguchi particle swarm optimization," *Eighth Int'l Conf. on Intelligent Systems Design and Applications*, vol. 2, pp. 307-312, 2008.
- Mhamdi, B., Grayaa, K. and Aguilu, T. "Hybrid of Particle Swarm Optimization, Simulated Annealing and Tabu Search for the Reconstruction of Two-Dimensional Targets from Laboratory-Controlled Data," *Progress In Electromagnetics Research B*, vol. 28, pp. 1-18, 2011.
- Mori H., Oshima K., and Nakamura M., "Generating avatar's facial expressions from emotional states in daily conversation," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5720-5723, 2011.
- Mpiperis, I., Malassiotis, S., Petridis, V. and M. Strintzis, G., "3D facial expression recognition using swarm intelligence," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2133-2136, 2008.
- OGRE – Open Source 3D Graphics Engine, 2012, <http://www.ogre3d.org>.
- Puklavage, C., Pirelas, A.R., Gonzalez, A.J. and Georgiopoulos, M., "Imitating Personalized Expressions in an Avatar through Machine Learning", *Florida AI Research Conf.*, May 2010.
- Thangaraj, R., Pant, M., Ajith, A. and Pascal, B., "Particle Swarm Optimization: Hybridization Perspectives and Experimental Illustrations," *Applied Mathematics and Computation*, 217(12), pp. 5208-5226, 2011.
- Velusamy, S., Kannan, H., Anand, B., Sharma, A. and Navathe, B., "A method to infer emotions from facial action units," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2028-2031, 2011.
- Yang S. and Bhanu B., "Facial expression recognition using emotion avatar image," *IEEE Int'l Conference on Automatic Face & Gesture Recognition and Workshops*, pp. 866-871, 2011.
- Zalewski L. and Gong, S. "2D statistical models of facial expressions for realistic 3D avatar animation," *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 217-222, 2005.
- Zhang Y. and Wu, L., "A Hybrid TS-PSO Optimization Algorithm," *Journal of Convergence Information Technology*, 6(5), pp. 169-174, 2011.
- Zhang Y. and Wu, L., "Tabu Search Particle Swarm Optimization used in Cluster Analysis," *J. of Science*, 1(1), pp. 6-12, 2012.