Journal of Visualization, Vol. 12, No. 3 (2009) 217-232

Regular Paper

Particle Tracking Velocimetry Using the Genetic Algorithm

Ohmi, K. *[1] and Panday, S. P. *[2]

*1 Faculty of Engineering, Osaka Sangyo University, Daito-shi, Osaka 5748530, Japan.
Tel 072-875-3001 Fax 072-870-1401 E-mail: ohmi@ise.osaka-sandai.ac.jp
*2 Graduate Student of Faculty of Engineering, Osaka Sangyo University, Daito-shi, Osaka 5748530, Japan.
Tel 072-875-3001 Fax 072-870-1401 E-mail: sanjeeb@wind.ise.osaka-sandai.ac.jp

Received 30 December 2008
Revised 16 March 2009

Abstract: A new concept genetic algorithm (GA) has been implemented and tested for the use in the 2-D and 3-D Particle Tracking Velocimetry (PTV). The algorithm is applicable to particle images with larger (greater than 2000) number of particles without losing the excellent accuracy in the particle matching results. This is mainly due to a new fitness function as well as unique genetic operations devised especially for the purpose of particle matching problem. The new fitness function is based on the relaxation of movement of a group of particles and is particularly suited for an increased density of particle images. The unique genetic operations give rise to the concentration of more fit genes in the forward part of the gene strings where the crossover and mutation processes are suppressed. The new algorithm also profits from the new genetic encoding scheme which can deal with the loss-of-pair particles (i.e., those particles which exist in one frame but do not have their matching pair in the other frame), a typical problem in the real image particle tracking velocimetry. In the present study, the new method is tested with 2-D and 3-D synthetic as well as real particle images with a large number of particles.

Keywords: Particle tracking velocimetry, 2-D PTV, 3-D PTV, Particle matching problem, Genetic algorithm.

1. Introduction

Particle Image Velocimetry (PIV) has become widely accepted as a reliable field measurement technique for the determination of velocity fields in the recent years (Adrian, 2004). PIV is a valuable optical diagnostic tool used to study fluids flows. It is a planar measurement technique wherein a pulsed laser light sheet is used to illuminate a flow field seeded with tracer particles small enough to accurately follow the flow.

Basically, there are three types of data processing techniques used in PIV: auto-correlation, cross-correlation and particle tracking. Correlation based processing techniques produce spatially averaged velocity estimates. The recorded image frame is divided into small sub-regions, each containing particle images. By processing the image over a regular grid of small sub-regions, a velocity vector map is generated. In contrast to the spatially averaged correlation techniques, Particle Tracking Velocimetry (PTV) techniques attempt to identify the displacement of individual particles. Both the techniques are equally important for the development of robust PIV algorithm. Particle tracking by itself is typically not capable of successfully tracking particles at the very high seed particle densities normally used for auto- or cross-correlation analysis. Conversely, correlation techniques must use large sub region sizes, with a concomitant reduction in spatial resolution, in order to perform adequately in the low seed particle density regimes where particle tracking techniques are normally applied. A hybrid system combining both the techniques can work well for the development of high-resolution PIV algorithm but either of these techniques needs a reliable and powerful algorithm.

From such a background, the most classical algorithms for PTV would be the four-frame in-line tracking method (Kobayashi et al., 1989; Hwang et al., 2005; Murai et al., 2008) using four consecutive particle image frames, and the binary-image cross correlation method (Uemura et al., 1989) using two frames instead of four. In the four-frame tracking method, the motion of tracer particles is traced frame by frame while considering the geometrical consistency of the resultant particle path. The advantage is that the method is simple in algorithm and applicable to particle images with rotating and/or shearing motion. But the method is usually applicable to low-density particle images only and the rate of velocity recovery is relatively low with respect to the number of particles. In contrast, the binary-image cross-correlation method is considered as a simplification of the standard cross correlation PIV (Raffel et al., 1998), in which the image correlation is computed for each interrogation window centered on the first-frame particle and that centered on the second-frame candidate particle. The computation is extremely quick because the cross-correlation function becomes a simple algebra with products of logical variables. The advantage and disadvantages of this method are in nearly inverse relation to those of the four-frame method; the speed and the velocity recovery ratio are improved but the cross-correlation algorithm, in principle, cannot be applied to strongly rotating and/or shearing flow images as in this method the parallel motion of the cluster particles is assumed.

In order to avoid this limitation, a couple of improved two-frame algorithms like the spring model by Okamoto et al. (1995), the affine transform by Okamoto (1998), and the velocity gradient tensor (VGT) by Ishikawa et al. (1997) have been proposed. All of these methods use a concept of particle cluster matching and start from the postulate that they should be applicable to non-linear deformation of flow patterns. However, most of these cluster based methods suffer from one serious ambiguity in their algorithms. A typical example of this is the spring model method. The concept of this method is very clear. All the particles of a cluster are interconnected by virtual springs. If these springs expand or contract between two images, the sum of spring forces is computed in each cluster. The combination of time-differential clusters which shows the least spring forces is a correct match pair of particle in the center of the clusters. In this method the spring force must be calculated from the Cartesian coordinates of neighbor particles in the cluster and in order to do so, the correct correspondence of neighbor particles must be known between the two time-differential clusters. But in the real flow images with non-linear deformation, this correspondence of neighbor particles is unclear in most cases and as a result, the spring force cannot be calculated.

The solution for this is to introduce an arbitrary assumption to this correspondence of neighbor particles. But the proposers of the spring model method have seemingly never made a detailed description of this assumption in the literature though the performance of the particle tracking is highly dependent on the method of assigning correspondence to neighbor particles. According to the present authors test results, what is even more worse is that the optimal method of assigning correspondence may be variable according to the flow conditions. Similar ambiguity is existent also in the affine transform and VGT based particle tracking.

In this context, there is one more skillfully conceived cluster-based method, namely the relaxation method (Baek and Lee, 1996; Ohmi and Li, 2000; Zhang et al., 2007) which is free from this ambiguity. This is indeed a good method with a concept of repeated updates of particle match probability but from the viewpoint of usability for non-experts of PTV experiments, the algorithm has to go with too many computational parameters.

Another new idea for particle tracking is the use of various kinds of cost functions, most of which are supposed to work as two-frame methods. A typical example is the multi-layer neural network application by Grant and Pan (1995). Although their algorithm is designed for double-exposure single frame particle images, the basic concept can be extended also for the use in the case of single-exposure image pairs. But one drawback of this type of neural network is that the system requires preliminary learning on the flow field to be estimated. Another example of the use of cost functions is the Hopfield neural network PTV implemented by Lee et al. (1995) and Knaak et al. (1997). This is indeed an interesting attempt with a concept of the minimization of Liapunov energy function but in order to get reasonable matching results, the energy function must be composed of four object functions representing the physical constraints of the flow. This often complicate the computation process with an additional problem of the weight balance of the multiple object functions.

By contrast, the self-organizing maps (SOM) neural network PTV proposed by Labonté (1999) is much simpler algorithm with less number of computational parameters to be tuned. This algorithm works well with relatively small number of particles but the matching result degrades rapidly at larger numbers (say more than 500) of particles. In addition, the degradation is more noticeable if the two images include more loss-of-pair particles. The new improved SOM algorithm by Ohmi (2003) works more effectively than the original Labonté's algorithm in many aspects despite the slightly increased complexity. One drawback of the new improved algorithm would be the re-addition of computational parameters to be tuned. The number of iteration cycles after which the output-layer neurons should be removed or doubled is a somewhat delicate parameter depending on the pattern of particle distribution.

If compared to all above cost function approaches, the genetic algorithm may be more directly and attractively applied to the particle matching process of PTV because the algorithm works according to a clearer concept of optimization inspired by the Darwinian evolutionary theory. Once the goal of the optimization is determined, the rest to be considered is all the technical problems concerning the tuning of genetic operations. From the viewpoint of the flow measurement, the genetic algorithm PTV has a remarkable advantage in comparison to many of other approaches: the algorithm goes without any *'a priori'* knowledge of flow to be measured. This means that the input of flow speed parameters (e.g. minimal and/or maximum flow rate) is not necessary unlike the other approaches.

However, the earlier genetic algorithm PTV has some drawbacks from the viewpoint of practicability. For instance, the number of trackable particles in a single run of computation is usually less than 100 or 200 at best. In addition, the existence of loss-of-pair particles present inevitably in real experimental images is not properly taken into account in the earlier algorithm. So in the present study, a new type of genetic algorithm implementation is attempted to overcome all these drawbacks of the earlier versions and to make them a more practically feasible PTV algorithm.

2. Genetic Algorithm PTV

The genetic algorithm has been already applied to 2-D particle tracking velocimetry by a number of authors e.g. Doh et al. (2001), Ohyama & Kaneko (1997) and Sheng & Meng (1998). The results are not bad at all as long as the particle density is relatively low and the mean particle displacement between the two frames is less than the mean particle interval in a single frame. However, they become often unsuccessful when the number of particles is increased and the mean particle displacement and the mean particle interval come close to each other. This limitation often comes from the use of a fitness function aiming at the least sum of the particle distances to be tracked. So for more robust algorithm, the fitness function should be based on some other criterion than the least sum of the particle distances.

In this regard, Kimura et al. (1998) and Furukawa et al. (1999) have proposed new fitness functions based on the morphology of neighbor particles centered on the particle to be tracked. As a result, the number of trackable particles is increased to a certain extent (three to four times) but never exceeds the order of 1000. One of the present authors and his co-worker (Ohmi and Yoshida,

2000) have further developed this idea to define a new fitness function based on the rigidity of the cluster pattern of particles. And as will be shown later in this work, this new fitness function now enables to track more than 2000 particles in relatively high-density particle images.

The computation time is usually considered as something specific to the genetic algorithm because it simulates basically the genetic evolution process of the animate nature. But as shown in the recent development of genetic manipulation of the farm animals and plants, it seems quite feasible also in the genetic algorithm to generate 'desirable' species in a relatively short cycle of generations. The new genetic operation (or manipulation) introduced in the present study is the individual gene sort according to the fitness index of the gene itself. This manipulation, followed by the suppression of crossover and mutation in the higher fit part of the gene strings, is very effective for accelerating the convergence of the genetic computation.

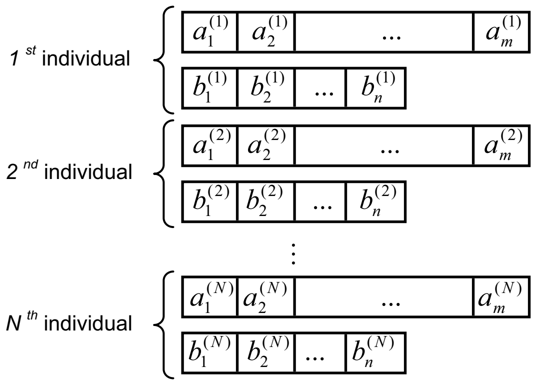
The treatment of loss-of-pair particles is mainly related to the genetic encoding scheme of the particle matching problem. In the encoding scheme by Furukawa et al. (1999), Kimura et al. (1998) and Ohmi & Yoshida (2000), it is supposed that every first-frame particle should find out its unique partner in the second frame so that the number of particles must be same between the two frames. In the algorithms by Ohyama & Kaneko (1997) and Sheng & Meng (1998), this restriction in the encoding scheme is slightly relaxed and different numbers of particles can be existent in the crossover and mutation steps. The latter authors, in particular, even developed a unique idea of detecting loss-of-pair particles through additional computation of generation cycles. But all of the above schemes are basically bijective functions and the inclusion of loss-of-pair particles is not explicitly allowed there. So in the present study, a new type of genetic encoding scheme has been devised in order to allow the existence of different numbers of particles in the two image frames without increasing (or even reducing) the generation cycles to be computed for their detection.

3. Genetic Encoding

The first step of the genetic algorithm is the encoding of the physical parameters in the form of genetic chromosomes. In the real PTV experiment, the particle images are recorded by an opto-electronic imaging process and the centroid coordinates of individual particles are obtained through a binarization and labeling computation process. In this latter process, the individual particles are assigned their own unique labels or ID numbers. And in the genetic encoding of the particle matching problem, this notion of particle ID is widely used in the earlier as well as in the present approaches.

The usual genetic encoding scheme for the particle matching problem uses a single gene string (chromosome), representing a sequence of second-frame particle IDs which should be matched with the first-frame particle IDs supposedly in serial order (1,2,3….). And in the initial stage, a number of individuals consisting of a single gene string with random order IDs are generated as the first generation parents. The gene string of every individual is then manipulated through iterative genetic operations, such as selection/reproduction, crossover and mutation, producing progressively such younger generations that should fit more to the goal of the fitness function. In the end, the gene string of the individuals shows a correct sequence of second-frame particle IDs corresponding to the first-frame particle IDs in serial order.

When the number of particles is different between the two frames, this single-string encoding scheme can be used with a simple reservation as described by Sheng & Meng (1998). The frame with a larger number of particles should be regarded as the second frame and the other frame as the first frame, the genetic operations are then iterated as in the case of the equi-numeral particle matching. Different rules may be applied in the initial coding and mutation processes, where the second frame particle IDs is in excess of the maximum first frame particles IDs. This rule modification may be easily implemented in the conventional GA particle tracking algorithm but obviously this type of encoding scheme is more likely to detect loss-of-pair particles only in the second frame and those in the first frame will be rather difficult to detect. Another problem of this simple rule modification is that the encoding system by itself does not allow the bi-directional particle matching (i.e. from the first to second as well as from the second to first), which is usually considered as an effective method of increasing the matching accuracy.


1 st individual
$a_1^{(1)}$ $a_2^{(1)}$ ... $a_m^{(1)}$
$b_1^{(1)}$ $b_2^{(1)}$ ... $b_n^{(1)}$
2 nd individual
$a_1^{(2)}$ $a_2^{(2)}$ ... $a_m^{(2)}$
$b_1^{(2)}$ $b_2^{(2)}$ ... $b_n^{(2)}$
N th individual
$a_1^{(N)}$ $a_2^{(N)}$ ... $a_m^{(N)}$
$b_1^{(N)}$ $b_2^{(N)}$ ... $b_n^{(N)}$

N: Number of individuals
m: Number of particles in the first frame
n: Number of particles in the second frame

Fig. 1. Genetic encoding scheme

Considering and taking all these problems and some other additional factors (described later) into account, the present authors propose here a new genetic encoding scheme as shown in Fig. 1. One unique point of the present genetic encoding is that each individual consists of two gene strings (chromosomes) representing the first-frame and second-frame particle IDs. And each pair of gene codes picked up from the same position of the first and second gene strings stands for the current matching result of particle IDs. In this encoding scheme, the bi-directional particle matching can be easily implemented by alternately changing the roles of the first-frame and second-frame particles in the iterative computation, i.e., the roles of the first-frame and second-frame particles are changed in the iterative computation after every two generations. This idea is schematically illustrated in Fig. 2. This bi-directional matching method together with the two chromosomes genetic encoding is clearly a more convenient strategy for detecting loss-of-pair particles in both the first and second frames.

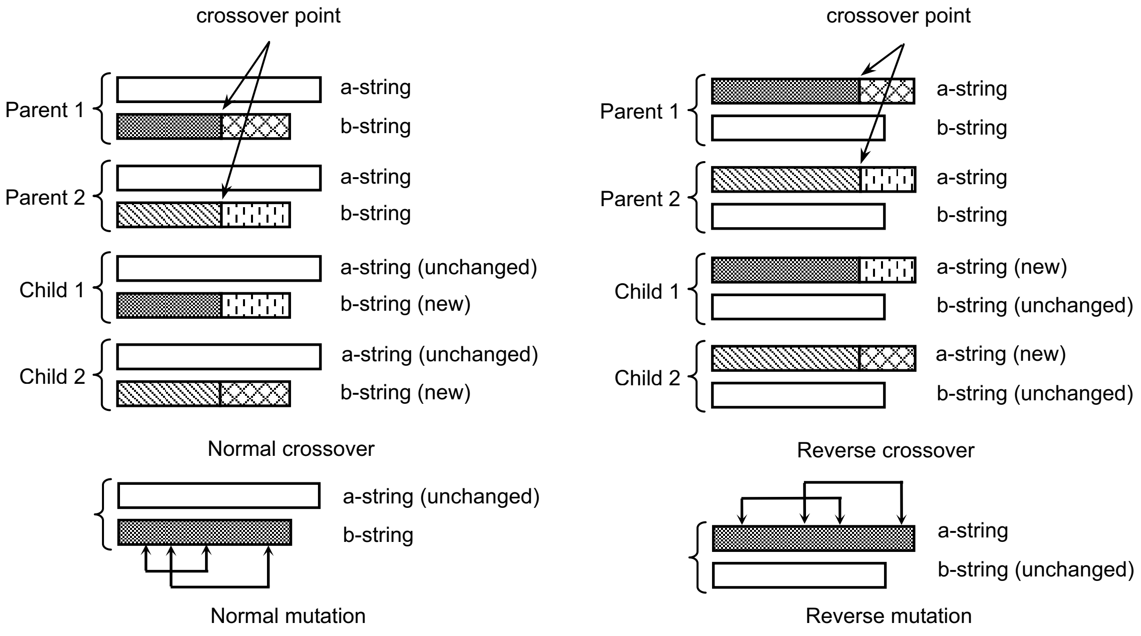


Fig. 2. Bi-directional particle matching

The two chromosomes (two gene strings) in this encoding scheme, however, causes genes to double in number compared to those of the conventional GA-PTV methods which causes the proposed GA-PTV to take slightly more time to converge than the conventional GA-PTV methods. But still when tested (with 2000 particles), the computation time is within tolerable limit. In addition, due to this encoding scheme, the proposed GA-PTV as will be shown in Section 6.2 later can be applied to the experimental particle images (real images) where there are loss-of-pair particles in both the first and second frames. This was not possible with conventional GA-PTV methods. Hence the authors feel that the use of this encoding scheme is much more beneficial in spite of slightly larger computation time.

Furthermore, using this new encoding scheme, the first frame gene string and the second frame one do not need to be exchanged according to their respective numbers of particles. The first gene string is simply regarded as the first frame particles and so is the second gene string. Also in the genetic operations, any rule modification as in Sheng & Meng (1998) is not necessary in the present encoding scheme.

4. Fitness Function

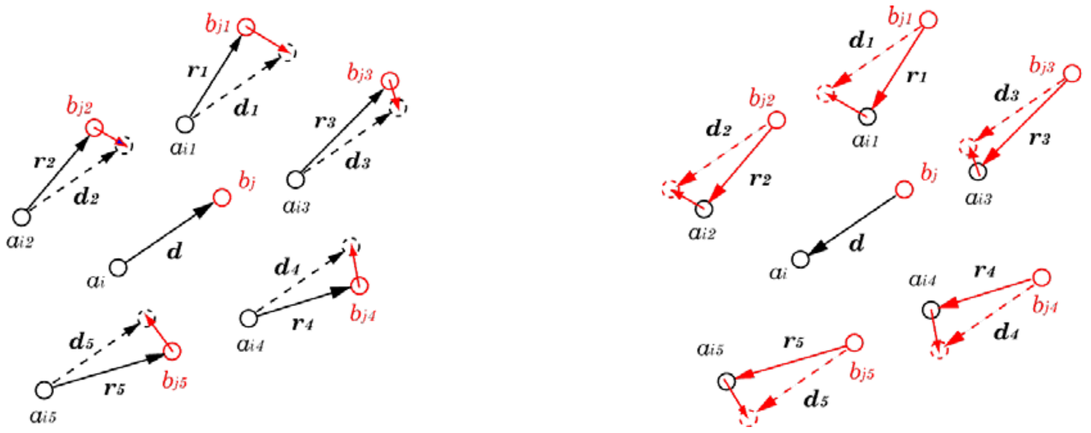
The fitness function is one of the key factors of the genetic algorithm particle tracking. As mentioned above, the most widely-used form of the fitness function uses the sum of the Euclidean distances of all the matched particle pairs, which should be minimized in the course of the genetic computation. The main drawback of this type of fitness function is that it cannot be successfully applied to the increased density particle images where the particles are located closely to each other. The situation is almost critical when two particles in a close proximity are traveling nearly in parallel, because the sum of the two particle travel distances may be shorter in the incorrect matching result than in the correct one. In order to avoid situations of this kind, the fitness function has to take into account not only the movement of the particle to be tracked but also that of its neighbor particles. And among a number of new ideas in such a direction, the fitness function based on the rigidity of the cluster pattern of neighbor particles (Ohmi and Yoshida, 2000) seems highly successful for most of practical use applications.

More precisely, this fitness function is defined as the sum of the relaxation lengths of all the neighbor particles. And this relaxation length is evaluated by the modulus of the difference vector between the virtual displacement vector parallel to the motion of the central reference particle and the real displacement vector nearest to the virtual one. Fig. 3 (a) illustrates the concept of this relaxation based fitness function, where the virtual displacement vector parallel to the motion of the central reference particle (from a_i to b_j) is denoted as $\boldsymbol{d_k}$ ($1 < k < p$, p is the number of neighbor particles) and the real displacement vector nearest to the virtual one (from a_{ik} to b_{jk}) is expressed as $\boldsymbol{r_k}$. Using these notations, the fitness function is given by:

$$F = \sum_{i=1}^{m} \sum_{k=1}^{p} \left| \boldsymbol{d}_k - \boldsymbol{r}_k \right| \tag{1}$$

where m stands for the number of particles in the first image frame.

In the case of the bi-directional particle matching, another fitness function in the reverse direction should be used at every two iteration cycles. The graphical explanation of this second fitness evaluation is given in Fig. 3 (b).



(a) Relaxation of normal particle matching

(b) Relaxation of reverse particle matching

Fig. 3. Concept of fitness function based on the sum of the relaxation lengths

5. Genetic Operations

The basic genetic operations in the present algorithm are not so different from what is called simple GA, except that the gene string (chromosome) is composed of a sequence of integer numbers instead of series of binary bits. After the initial population (initial group of individuals) is generated, the individuals go through the iterative cycle consisting of selection/reproduction, crossover and mutation. But in order to converge the computation results more quickly, a couple of new ideas are introduced in the conventional genetic operations. Needless to say, the new ideas are aimed at the particle matching problem and may not necessarily be applicable to other types of optimization problems.

5.1 Generation of Intitial Population

The individuals of the initial population are composed of two gene strings representing random matches of particle IDs between the first and second frames. Most typically, the first-frame gene string (chromosome) is assigned a sequence of serial integer numbers (from 1 to m), and the second-frame gene string is given a series of random integer numbers (between 1 and n). The initial population is therefore composed of individuals with two different gene string lengths and this structure of the individuals is retained until the end of the iterative computation.

5.2 Selection and Reproduction

The fitness of each individual is evaluated in terms of the fitness function given in equation (1). And individuals with more fitness have more possibility of surviving in the next generation. The genetic operation simulating this biological process is often called as selection/reproduction or simply selection. There are two selection methods used in the present study; the first one is the exponentially ranking selection, in which the numbers of individuals to be reproduced are determined by the exponential ranking of their fitness evaluation. The second one is the unconditional elite-reproducing selection, in which the best-fit individual in the generation history is unconditionally reproduced with a prescribed rate of reproduction. In this latter selection method, the reproduction rate is progressively decreased with the growth of generation in order to avoid the wandering of the best fitness around a local optimum.

Between these two selection methods, the elite-reproducing selection is generally more powerful and effective for quick convergence, unless the gene code sort operation (described in the next section) is introduced. If the gene code sort operation is present, the choice of these two selection methods depends on circumstances because both of these methods work quite effectively if employed with some specific methods of crossover and mutation.

5.3 Sort of Gene Strings

After the selection/reproduction process the gene code pairs on the two chromosomes of each individual is sorted according to the fitness index of the gene code (i.e. particle ID) pair. This fitness index is usually computed not only by the fitness function (1) without summation of i but also by the Euclidian particle distance at every 10 to 50 generations. The objective of this sort is to collect more fit (or presumably more probable) gene code pairs in the forward part of the two gene strings, where the crossover and mutation are purposely suppressed in the next computation step. In this way, more fit gene code pairs have more possibility of being protected from destruction by the crossover and mutation and, as a result, the computation comes to convergence very quickly. This is really a unique point of the present genetic algorithm, probably never seen before in the simple GA.

5.4 Crossover

The next genetic operation is the crossover. In the present study, two crossover methods are used according to circumstances. The first one is the multi-point inversion crossover in Fig. 4 and the second one is the single-point partially matched crossover in Fig. 5. The former crossover method, one of the unique ideas of the present study, is considered as a combination of the multi-point crossover and the gene-string inversion scheme. In this method, any doubled gene code of one child string is exchanged with that of the other child string in the order of their appearance. The latter one, also known as PMX in abbreviated form, is a relatively often used crossover method in the GA models with non-binary gene codes.

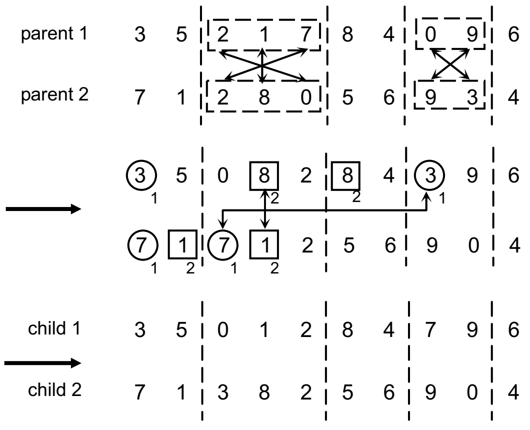


Fig. 4. Multi-point inverse crossover

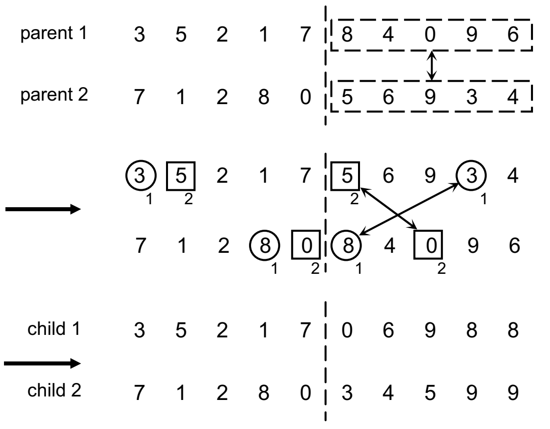


Fig. 5. Single-point partially matched crossover

5.5 Mutation

In the present mutation method, randomly selected two gene codes in a single gene string are exchanged as illustrated in Fig. 6. The mutation rate is constant through the iterative computation as long as the gene code sort operation is applied.

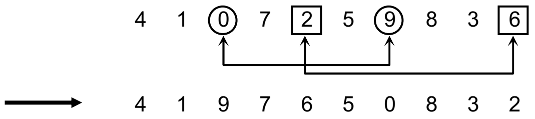


Fig. 6. Random gene-exchange mutation

As all the above genetic operations are iterated, the fitness function (1) is converged at a constant value and both of the two gene strings in the individuals come to indicate invariable and fixed gene code pairs. These gene code pairs are taken as the final particle matching results.

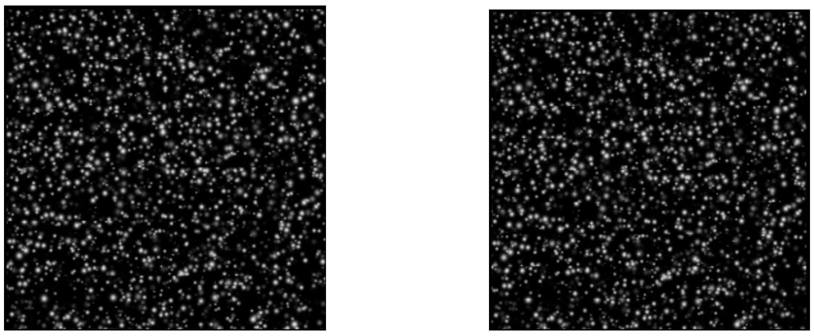
6. Results

6.1 Synthetic Particle Image

The present particle tracking algorithm is first applied to the PIV standard image (Okamoto et al., 2000a) published by the Visualization Society of Japan (VSJ). This is the generic name of various series of synthetic particle images based on the Cumulative Frequency Distribution (CFD) results of a 3-D impinging jet in a square cavity. There are many series of particle images derived from different portions of the flow filed, which are all available through the Internet (http://www.vsj.or.jp/piv). One of the advantages of this PIV standard image is that the images come with the original particle coordinates data, so that one can compare his own particle tracking results with the correct coordinates data.

Most of the images are composed of four consecutive frames with 256 by 256 pixels resolution. The particles are distributed at random and so are the particle coordinates data in the accompanying data set. Therefore, if this data set is read from the beginning, the resultant particle coordinates are also distributed at random regardless of the number of read records. The particle tracking algorithm is tested by this type of particle coordinates data, while the number of read particles is specified in each trial.

Here, the tested PIV standard image is listed as series #301, from which the first and second frames are used. The first two exposures of this set of particle images are shown in Fig. 7. Between these two frames, the particle displacement is 10 pixels at maximum. The number of existing particle pairs is around 4000, which is almost the limit for the present genetic algorithm to be converged in realistic computing time.

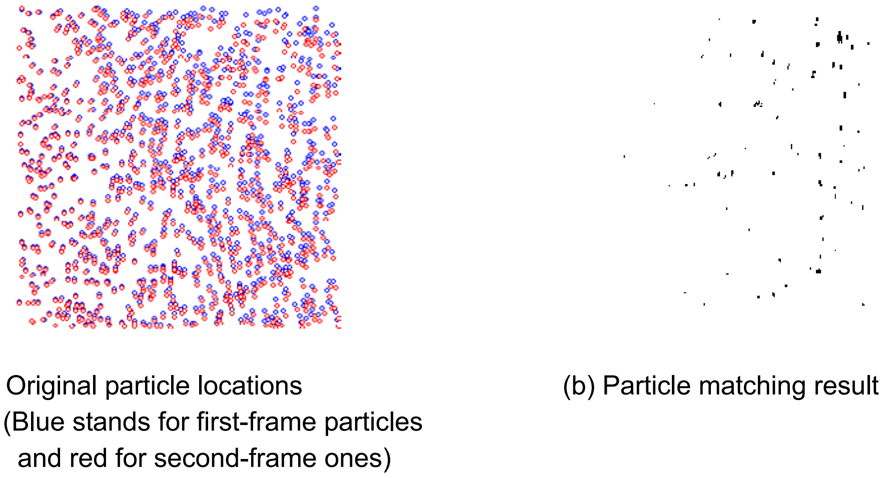


(a) First exposure image (b) Second exposure image

Fig. 7. PIV standard image #301 (with 1/30 s time interval)

Fig. 8 shows the result of particle tracking for 1000 particles compared with their original particle data. The major parameters employed in this test are as follows: the number of individuals is

10, maximum generation 4000, number of neighbor particles 6, crossover rate 1 (the multi-point inversion crossover), mutation rate 0.2 and the base of the exponent is 0.2 (exponentially ranking selection). The computation time on a 1.8 GHz Core2Duo PC was around 64 seconds and the correct matching rate was 100% (without any error) obtained in about 1535 number of generations.

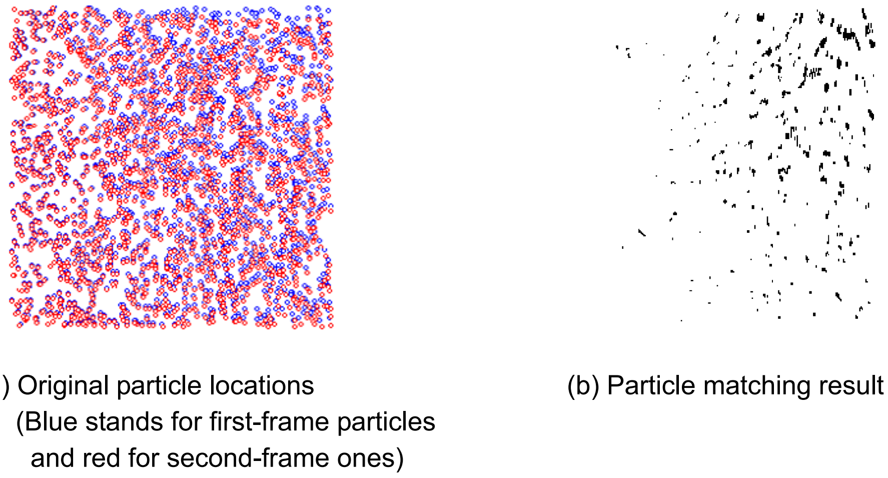


(a) Original particle locations
(Blue stands for first-frame particles
and red for second-frame ones)

(b) Particle matching result

Fig. 8. Particle tracking result from the data set with 1000 particles

Fig. 9 shows the result for 2000 particles compared with the original particle data. The major parameters employed in this case are similar to the one employed for 1000 particles above. The computation time on the same PC as mentioned above was around 535 seconds and the correct matching rate was also 100% obtained in about 3333 number of generations.



(a) Original particle locations
(Blue stands for first-frame particles
and red for second-frame ones)

(b) Particle matching result

Fig. 9. Particle tracking result from the data set with 2000 particles

All the parameters employed here are determined on a cut-and-try basis. In most of GA-based PTV methods, the algorithm does not require any preset parameters derived from the physical flow conditions to be examined. This is one of the prominent advantages of the method. But the reverse side of this advantage is that there are no physical or theoretical backgrounds by which the parameters are determined automatically. The only parameter related to the flow condition in the present version of GA-PTV would be the number of neighbor particles (more precisely the distributed area) in the clusters. Bearing this in consideration, the dependency of the matching rate on the number of neighbor particles are presented in Table 1 for 2000 particles.

Table 1. Dependency of the matching rate on the number of neighbor particles

Number of Particles	Number of Generations	Number of Neighbor Particles	Correct Matching Rate (%)	Computation Time (s)
2000	4000	4	99.8	665
2000	3333	6	100.0	535
2000	3148	8	100.0	541
2000	3074	10	100.0	546
2000	3020	12	100.0	568

It can be clearly observed from Table 1 that if we increase the number of neighbor particles above 6, the optimum neighbor particles in this case, the computation time increases, i.e., the algorithm requires more time to converge to the solution even though the number of generations are less. Below 6, the algorithm requires more generation as well as more time to converge to the optimum solution.

6.2 Experimental Particle Image

The second particle tracking test is conducted by using particle images derived from a real flow experiment. The test image used here is called as "PIV benchmark test image" and derived from the flow visualization around an oscillating flat plate in still water. Every detail of this test image is given by Hayami et al. (1997). The test image consists of four consecutive frames which are rather noisy and narrow-band in histogram. In addition, the mean gray level is fluctuant from frame to frame. This is why the benchmark test image is a challenge for every PIV algorithm including that of the present work.

The first goal of this benchmark test image for PTV approaches is the detection of individual particles. In the present study, the individual particles are found by the dynamic threshold binarization by Ohmi & Li (2000), according to which the numbers of particles are 1830 and 1893 in the first and second frames respectively. These two frames with maximal particle displacement of 7 pixels are used for the particle tracking test. Fig. 10 shows the overlap of two time-differential PIV benchmark test images with 512 X 512 pixels resolution and 8 bit/pixel quantization. There are loss-of-pair particles or the particles not likely to find their partners in the opposite frame, as indicated by isolated red and blue closed circles in Fig. 11 (a). Fig. 11 (b) shows the result of particle matching from this pair of images.

The major parameters employed in Fig. 11 (b) are as follows: the number of individuals is 10, maximum generation 15000, number of neighbor particles 8, crossover rate 1 (single-point partially matched crossover), mutation rate 1 and the reproduction rate is 0.2 (elite-reproducing selection). The computation time on a 1.8 GHz Core2Duo PC was around 3337 seconds for establishing 1583 particle pairs. Fig. 11 (b) shows only these 'correct' matching results. This obviously indicates that the genetic algorithm PTV works effectively and successfully even in the case of rather difficult experimental images.

The computation time can be considerably reduced, if a certain threshold is introduced for the fitness index of the chromosome gene pair *and* the unconditional protection strategy is applied to the gene pairs below this threshold. However, the use of this additional threshold parameter may be considered as a loss of the significant merit of the genetic algorithm, because this parameter is closely related to the physical quantities of the flow to be examined. In other words, the genetic algorithm PTV may also be dependent on *'a priori'* knowledge of the flow field.

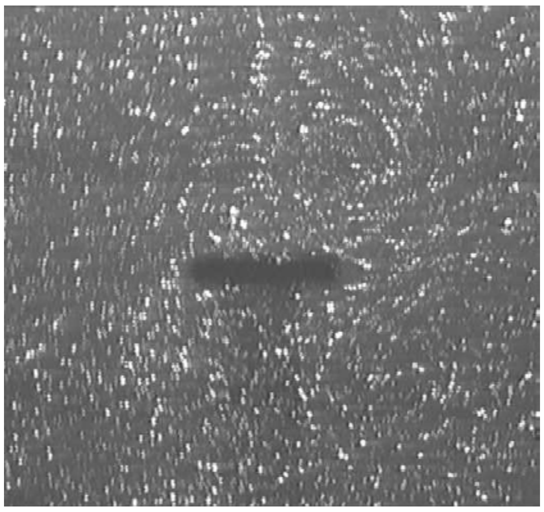
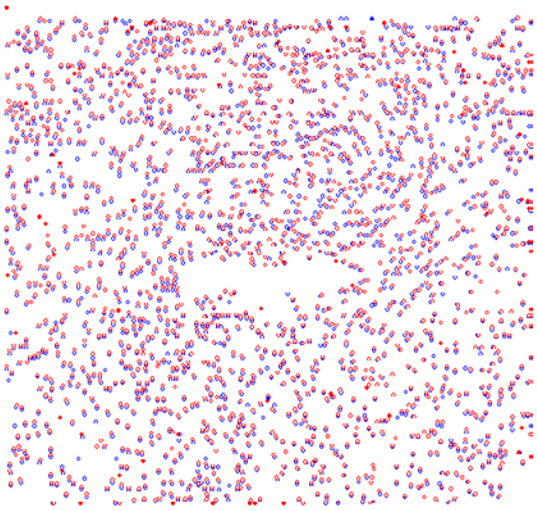


Fig. 10. PIV benchmark test image (overlap of two-differential frames)



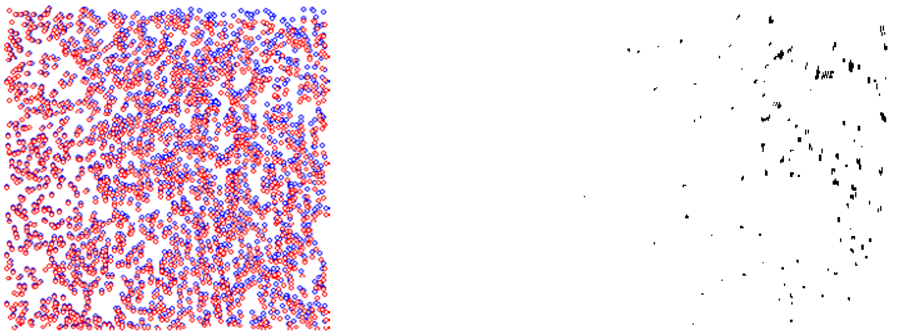
(a) Original particle locations (Blue and red open circles stand for paired first-frame and second-frame particles respectively; Blue and red closed circles are loss-of-pair particles in first and second frame respectively, detected by the present GA particle tracking)

(b) Particle matching result

Fig. 11. Particle tracking result of the PIV benchmark test image

6.3 Comparison with Conventional GA-PTV method

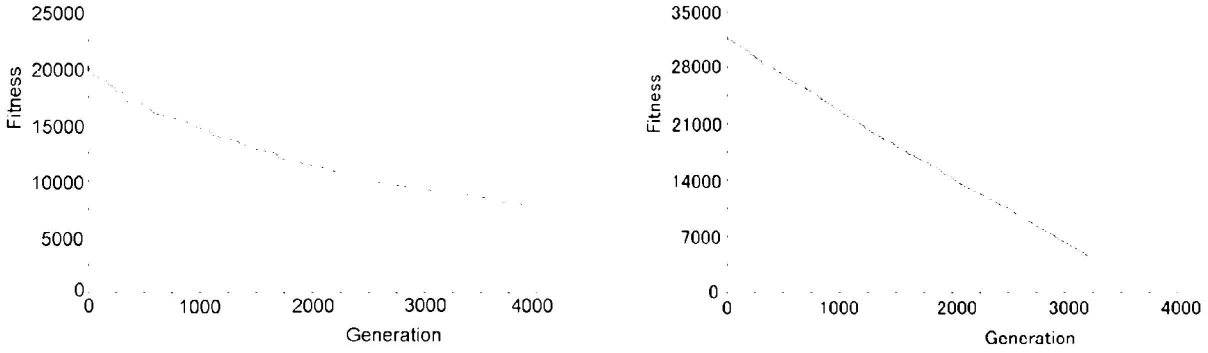
Kimura et al. (1998) have proposed fitness function based on morphology of neighbor particles centered on the particle to be tracked. This can be regarded as one of the conventional GA-PTVs in which the cluster including neighbor particles are used. The present authors have applied this fitness function (Kimura et al., 1998) for particle pairing using GA. Fig. 12 shows the result for 2000 particles compared with the original particle data using this function. The major parameters employed in this case are similar to the one employed for synthetic particle images previously in the section 6.1. The computation time on a 1.8 GHz Core2Duo PC was around 695 seconds and only 1295 particles were detected out of 2000 particles in 4000 number of generations. The correct matching rate was only 64.8% as compared to 100% obtained in 535 seconds and 3333 number of generations by proposed GA-PTV. This result clearly shows that the performance of the proposed GA-PTV is much better than the conventional GA-PTV (Kimura et al., 1998). Fig. 13 (a) and Fig. 13 (b) show the graph between the fitness function and number of generations for conventional GA-PTV (Kimura et al., 1998) and proposed GA-PTV respectively. Comparing these two figures one can surely ascertain the effects of the unique genetic operators and the superiority of the fitness function (1) introduced in this paper.



(a) Original particle locations
(Blue stands for first-frame particles
and red for second-frame ones)

(b) Particle matching result

Fig. 12. Particle pairing result for 2000 particles using conventional GA-PTV (Kimura et al., 1998)



(a) Conventional GA-PTV

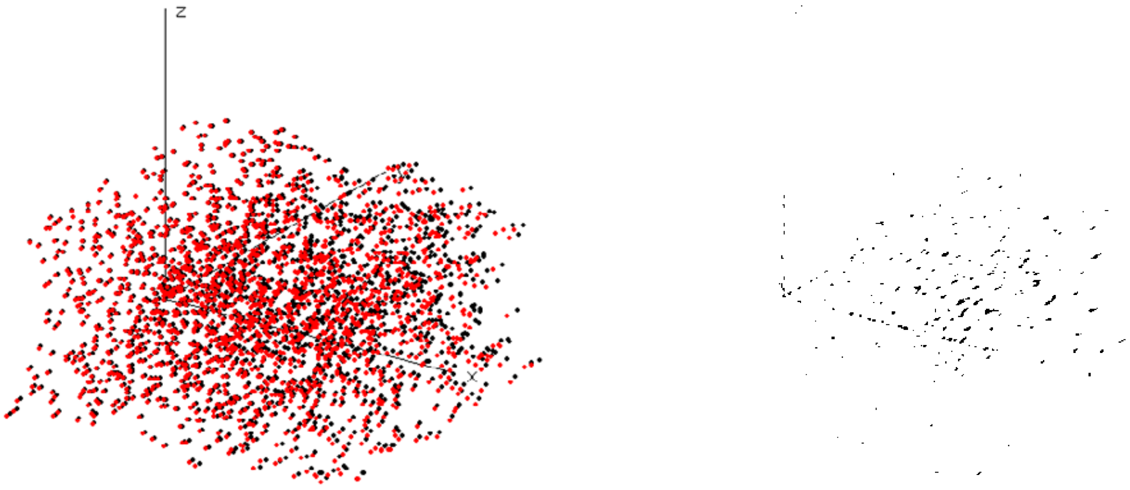
(b) Proposed GA-PTV

Fig. 13. Graph between the fitness function and number of generations

7. 3-D Particle Tracking

In the next step, the test of the 3-D particle tracking is conducted by using the library of the 3-D PIV Standard Image implemented by Okamoto et al. (2000b). The present GA based particle tracking algorithm is applied to the time-differential datasets of 3-D particle coordinates obtained directly from their numerical library as exact values.

The 3-D particle match results are compared to the known displacement data of the particles and thereby the performance of the particle tracking is evaluated. The major parameters employed in this case are similar to the one employed for synthetic particle images previously in the section 6.1. The result for 2000 particles compared with their original particle data is shown in Fig. 14. The computation time on a 1.8 GHz Core2Duo PC was around 474 seconds and the correct matching rate was 100% obtained in 2648 number of generations.



(a) Original particle locations
(Red dot stands for first-frame particles
and black dot stand for second-frame ones)

(b) Particle matching result

Fig. 14. Particle tracking results of 3-D PIV Standard Image #351 with 2000 particles

8. Conclusion

Genetic algorithm has been successfully applied to the particle matching process of the particle tracking velocimetry. Despite the start from completely random pairs of particles from the first and second image frames, the computation time remains in a reasonable to acceptable range with the help of a new genetic encoding scheme and fully worked-out genetic operations. The number of trackable particles is now more than 2000, while loss-of-pair particles can be existent in either of the two frames. Another concern of the present PTV algorithm would be the accuracy of particle tracking. In this connection, a benchmark test has been conducted regarding the correct matching rate of the particle tracking in the PIV standard image #301. The results are presented in Table 2 for various numbers of particles to be tracked.

Table 2. Correct matching rate and computation time of the particle tracking in the PIV standard image #301

Number of particles	Correct matching rate (%)	Computation time (s)
100	100.0	0.2
200	100.0	0.8
400	100.0	4.3
700	100.0	19.5
1000	100.0	64
2000	100.0	535
2200	99.7	718

The number of the relaxation neighbor particles is fixed at 6 or 8 in the present work and the correct matching rate could be improved furthermore with an increased number of neighbors. As shown in this table, a perfect matching has been obtained up to 2000 particles (though not constantly depending on the random numbers generated) but the rate is away from 100 % thereafter. Table 2 shows also the computation time for every number of particles. Although the computation time is a dependent factor and should not be an object of a general discussion, reasonable results are seen up to 2000 particles, which as it happens corresponds to the upper limit of the 100 % correct matching. Therefore, it can be concluded that the genetic algorithm has capability of tracking more than 2000 particles and the accuracy is not bad at all, though the best performance can be obtained when tracking 2000 particles or less. Furthermore, the relevant statistical data of 3-D PTV results presented in Table 3 clearly shows that the proposed GA-PTV has also been successfully applied for the 3-D PTV.

Table 3. Correct matching rate and computation time of the 3-D particle tracking in the PIV standard image #351

Number of particles	Correct matching rate (%)	Computation time (s)
2000	100.0	474

References

Adrian, R.J., Twenty Years of Particle Image Velocimetry, Proceedings of the 12th International Symposium on Applications of Laser Techniques to Fluid Mechanics, (2004), #01-1.

Baek, S.J. and Lee, S.J., A new two-frame particle tracking algorithm using match probability, Experiments in Fluids, 22-1 (1996), 23-32.

Doh, D. H., Kim, D. H., Cho, K. R., Cho, Y. B., Lee, W. J., Saga, T. and Kobayashi, T., Development of Genetic Algorithm Based 3D-PTV Technique, Journal of Visualization, 5-3 (2002), 243-254.

Furukawa, T., Kimura, I., Kuroe, Y. and Kaga, A., Hybrid PTV using Neural Networks and Genetic Algorithms, Proceedings of the 2nd Pacific Symposium on Flow Visualization and Image Processing, (1999), PF-075.

Grant, I. and Pan, X., An Investigation of the Performance of Multi-Layer Neural Networks Applied to the Analysis of PIV Images, Experiments in Fluids, 19-3 (1995), 159-166.

Hayami, H., Oakmoto, K. and Aramaki, S., A Trial of Benchmark Test for PIV (in Japanese), Journal of Visualization Society of Japan, 17-S1 (1997), 163-166.

Hwang, T. G., Doh, D. H. and Okamoto, K., 4D-PTV: Measurements of an Impinged Jet with a Dynamic 3D-PTV, Journal of Visualization, 8-3 (2005), 245-252.

Ishikawa, M., Yamamoto, F., Murai, Y., Iguchi, M., Wada, A., A novel PIV algorithm using velocity gradient tensor, Proceedings of the 2nd International Workshop on PIV'97-Fukui, (1997), 51-56.

Kimura, I., Hattori, A. and Ueda, M., Particle Pairing Using Genetic Algorithms for PIV, Proc. VSJ-SPIE 98, (1998), AB-093.

Knaak, M., Rothlübbers, C.and Orglmeister, R., A Hopfield Neural Network for Flow Field Computation Based on Particle Image Velocimetry / Particle Tracking Velocimetry Image Sequences, Proceedings of the IEEE International Conference on Neural Networks, (1997), 48-52.

Kobayashi, T., Saga, T. and Segawa, S., Multipoint Velocity Measurement for Unsteady Flow Field by Digital Image Processing, Flow Visualization V, Hemisphere, (1989), 197-202.

Labonté, G., A New Neural Network for Particle Tracking Velocimetry, Experiments in Fluids, 26-4 (1999), 340-346.

Lee, J., Principe, J.C. and Hanes, D.M., Velocity Measurement of Granular Flow with a Hopfield Network, Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, (1995), 380-387.

Murai, Y., Oishi, Y., Tasaka, Y. and Takeda, Y., Particle Tracking Velocimetry Applied for Fireworks: A Demonstration of Vector Field Measurement in Hundreds Meter Space, Journal of Visualization, 11-1 (2008), 63-70.

Ohmi, K. and Yoshida, N., A New Type of GA-Based Particle Tracking Velocimetry, Proceedings of 9th International Symposium on Flow Visualization, (2000), #401.

Ohmi, K. and Li, H., Particle Tracking Velocimetry with New Algorithms, Measurement Science and Technology, 11-6 (2000), 603-616.

Ohmi, K., 3-D Particle Tracking Velocimetry Using a SOM Neural Network, Proceedings of 5th International Symposium on Particle Image Velocimetry, (2003), #3112.

Ohyama, R. and Kaneko, K., Experimental Study on Space and Time Correspondence of Traveling Particles for Three-Dimensional Particle Image Velocimetry by Genetic Algorithm, Proc. SPIE, Vol.3172, (1997), 688-699.

Okamoto K., Schmidl W.D., Hassan Y.A. Least force technique for the particle tracking algorithm, Flow Visualization VII, Begell House, (1995), 647-652.

Okamoto K., Particle cluster tracking algorithm in particle image velocimetry, JSME International Journal, Series B, 41-1 (1998), 151-154.

Okamoto K., Nishio, S., Saga, T., Kobayashi, T., Standard images for particle image velocimetry, Meas. Sci. Technology, 11 (2001a), 685-691.

Okamoto, K., Nishio, S., Kobayashi, T., Saga, T., Takehara, K., Evaluation of the 3D-PIV Standard Images (PIV-STD Project), Journal of Visualization, 3-2 (2000b), 115-124.

Raffel, M., Willert, C.E. and Kompenhans, J., Particle Image Velocimetry, - A Practical Guide, (1998), Springer-Verlag, Heidelberg, Berlin.

Sheng, J. and Meng, H., A Genetic Algorithm Approach for 3D Velocity Field Extraction in Holographic Particle Image Velocimetry, Experiments in Fluids, 25 (1998), 461-473.

Uemura, T., Yamamoto, F. and Ohmi, K., High Speed Algorithm of Image Analysis for Real Time Measurement of Two-Dimensional Velocity Distribution, Flow Visualization: ASME FED-85, (1989), 129-134.

Zhang, W., Kang, J. H. and Lee, S. J., Visualization of Saltating Sand Particle Movement near a Flat Ground Surface, Journal of Visualization, 10-1 (2007), 39-46.

Author Profile

Kazuo Ohmi (Member) was born in Osaka, Japan. He's date of birth is September 19th, 1951. He received his M.Sc. in mechanical engineering from Osaka University, Japan in 1979. He received his Ph.D. in energetic from Universit´e de Poitiers, France in 1987. He also received doctoral degree in mechanical engineering from Osaka University in 1991.

He worked in Laboratoire de M´ecanique des Fluides, Universit´e de Poitiers, France, as a visiting researcher from 1984 to 1987. He is currently a professor in Department of Information Systems Engineering, Osaka Sangyo University, Japan. He was in Politecnico di Torino, Italy, as a visiting researcher from 1999 to 2000. His research interests are quantitative visualization, PIV, PTV, Holographic PIV, wakes and vortices, applied artificial intelligence, visualization of art and music and so on. Prof. Ohmi is a member of various professional societies including IEEE, IEICE, American Society of Mechanical Engineers, American Physical Society and Visualization Society of Japan etc.

Sanjeeb Prasad Panday (Student-member) was born in Kathmandu, Nepal. He's date of birth is June 4th, 1976. He received the Bachelors Degree in electrical engineering from University of Engineering and Technology, Lahore, Pakistan in 2001 and Masters Degree in information and communication engineering from Tribhuvan University, Nepal in 2006. He is currently pursuing a Doctoral Degree in information systems engineering at the graduate school of engineering, Osaka Sangyo University, Japan. He has been working as an assistant professor in the Department of Electronics and Computer Engineering at Institute of Engineering, Pulchowk Campus, Tribhuvan University, Pulchowk, Lalitpur, Nepal since 2002. He is currently on the study leave from Tribhuvan University for his higher studies. His research interests include image processing, digital holography, algorithms and their application to flow field measurements. Mr. Panday is now a member of Visualization Society of Japan and IEEE.