

PAPER

A Neural Network Based Algorithm for Particle Pairing Problem of PIV Measurements

Achyut SAPKOTA^{†a)}, *Student Member* and Kazuo OHMI^{†b)}, *Member*

SUMMARY Particle Image Velocimetry (PIV) is a widely used tool for the measurement of the different kinematic properties of the fluid flow. In this measurement technique, a pulsed laser light sheet is used to illuminate a flow field seeded with tracer particles and at each instance of illumination, the positions of the particles are recorded on digital CCD cameras. The resulting two camera frames can then be processed by various techniques to obtain the velocity vectors. One such techniques involve the tracking of the individual particles so as to identify the displacement of the every particles present in the flow field. The displacement of individual particles thus determined gives the velocity information if divided by known time interval. The accuracy as well as efficiency of such measurement systems depend upon the reliability of the algorithms to track those particles. In the present work, a cellular neural network based algorithm has been proposed. Performance test has been carried out using the standard flow images. It performs well in comparison to the existing algorithms in terms of reliability, accuracy and processing time.

key words: flow measurement, particle image velocimetry, particle tracking velocimetry, neural network, optimization

1. Introduction

Enhancements in the visualization of flow fields have been achieved in the recent years with the advancement of digital image processing techniques together with electronic and optical hardware. Particle Image Velocimetry (PIV) has been widely accepted as a reliable technique for the determination of velocity fields. In this measurement technique, a pulsed laser light sheet is used to illuminate a flow field seeded with tracer particles, and at each instance of illumination the positions of the particles are recorded on digital CCD cameras. The two camera frames are then processed to find the displacement vector map of the flow field. The displacement that can be estimated by different means and the time separation between the images give the velocity information. In contrast to the techniques for the measurement of flow velocities employing probes as pressure tubes or hotwires, the PIV technique being an optical technique works intrusively. This allows the application of PIV even in high-speed flows with shocks or in boundary layers close to the wall, where the flow may be disturbed by a presence of probe. This technique allows one to record images of large parts of flow fields in a variety of applications in gaseous and

liquid media and to extract the velocity information. This feature is unique to PIV as most of the traditional techniques for velocity measurements only allow the measurement of velocity of the flow at a single point [1].

Basically, there are three types of particle data processing techniques used in PIV: auto-correlation, cross-correlation and individual particle tracking, which is commonly known as Particle Tracking Velocimetry (PTV). Correlation based processing techniques produce spatially averaged velocity estimates while PTV technique attempts to identify the displacement of individual particles. The later one is the focus of the present study. The technique of individual particle tracking has two advantages over the correlation based processing techniques. The first one is the localization of the measured velocity within the limit of the traceability of particles. In the case of correlation based techniques, the aim is to calculate mean velocity of a group of particles. The possibility of taking fully three-dimensional measurements by the use of volumetric lighting and stereoscopic analysis is another advantage of the PTV over correlation based PIV techniques.

In PTV, all the particles in an image taken at one time step (say, at $t = t_n$) are compared to the particles in another image taken after certain time interval (say, at $t = t_{n+1}$) to determine physically correspondent particle pairs. The displacements of individual particles thus determined gives the velocity information if divided by a known time interval i.e. $t_{n+1} - t_n$. This kind of correspondence problem, in PTV measurements, has been dealt in different ways to date. The initial attempts were made using multi-frame tracking [2], [3] and binary image cross-correlation [4] techniques. The multi-frame tracking methods, which require four or more frames, involve the iterative procedures of extrapolation of particle displacement and search for the nearest neighbor. The key assumption behind this method is the small displacement of the particle from one frame to another. Only on the basis of this constraint, many ambiguous situations are obvious in the particle frames with higher particle density.

Similarly, binary image cross-correlation technique is a variation of standard cross-correlation particle image velocimetry and requires only two frames. This involves the computation of the cross-correlation between the regions around the particles in the first and in the second frame. The number of particles that can be matched accurately is more in comparison to multi-frame tracking methods mentioned earlier. The point which makes this algorithm less

Manuscript received March 11, 2008.

Manuscript revised September 2, 2008.

[†]The authors are with the Department of Information Systems Engineering, Osaka Sangyo University, Daito-shi, 574-8530 Japan.

a) E-mail: sapkota@ieee.org

b) E-mail: ohmi@ise.osaka-sandai.ac.jp

DOI: 10.1587/transinf.E92.D.319

reliable is the assumption of equal velocity of the particles within the cross-correlation window. The positive estimate of the correlation is increased only if the neighboring particles find their partners at exactly the same relative displacement points. It is not always the case and this is the reason why the algorithm is difficult to apply for the cases of strongly rotating and shearing motion.

Similarly, Wernet [5]–[8] and others [9]–[11], including the present authors, made assessments of the fuzzy logic based algorithm for particle tracking velocimetry. In these methods, each particle in the second frame within the user-defined radius from the position of the particle in the first frame is considered as candidate particles of this first frame particle. In the list of candidate particles thus generated, if two initial particles claim the same second frame particle then the pair of vectors that look the most similar (in direction and magnitude) must be correct pair of displacement vectors for these two separate initial particles. This algorithm is good in a sense that it tends to address the different constraints of the flow fields but the limitation here is requirement of the priori knowledge of the maximum flow velocity to search the probable candidates. The small size of the search region may cause the missing of the true vector in the set of interacting vectors and large value may result to more confusing interacting vectors especially in the region of lower velocity gradient and high particle density. Moreover, the method works under the supervision of the fuzzy rules which requires the careful judgment of the system. Similarly, the appropriate assignment of the membership values is another complicated part of this algorithm.

Another approach for the particle tracking algorithm is the matching of the particle clusters [12], [13]. In this type of particle tracking, each one of the first-frame and second-frame particles form a cluster together with its respective neighboring particles and the selection of the best-match partners is conducted on the basis of a deformation index defined for the relationship between the first-frame and second-frame clusters. These algorithms require a fixed number of neighboring particles and, in order to satisfy this condition, the size of the local interrogation area has to be changed from cluster to cluster. This causes the problem of the localization of velocity data. Similarly, the approach shares its limitation with fuzzy logic based method that the information of the maximum flow velocity is needed in advance to choose the candidates.

The relaxation algorithms [14]–[19] based on the probability of particle matching [20] seem promising in comparison to the above methods. In the relaxation algorithm, every first-frame particle selects its candidate partners from the second frame using a certain distance threshold. Similarly, every first frame particle selects its neighbors using again the distance threshold. The probabilities of matching between the first-frame particles and their respective candidates in the second frame are calculated. The particle probabilities are then updated using the matching probabilities of the neighboring particles which are then iterated until all the probabilities remain almost constant. The basic concept

of the relaxation method is somewhat similar to that of the binary-image cross-correlation method on the basis of the principle that every possible movement of the reference particle in the first frame is checked in the light of neighboring particles. Unlike binary-image cross-correlation method, the positive estimate in the relaxation method is increased only if the neighboring particles find their partners within the finite distance from the parallel displacement point. This makes the algorithm applicable to rotating or shearing motions. But, like fuzzy logic based algorithm, it also requires the priori knowledge of the flow field as distance thresholds are determined on the basis of the maximum possible displacement of the particles, and on the basis of the radius of the neighborhood where the similarity of the movement of the particles can be preserved. In addition, relaxation algorithm works on a fixed set of particles during its iterative process. Therefore, it requires in the initial step enough matching particles to stimulate the matching process and converge towards a reliable solution. Not satisfying this requirement makes difficult the convergence of the probability and may result in the situation where no match particles are found. This problem is more pronounced for the images with low particle density.

In these contexts, the authors believe that defining and optimizing a certain cost function representing the relation between the particles in the two image frames is the most suitable solution for the particle tracking methods. Such methods can go without priori knowledge of the flow and can accommodate the different conditions that the paired particles must satisfy. The genetic algorithms and neural networks are powerful tools for many types of optimization problems and thereby used as prospective methods in the particle pairing process of PTV. Ohyama et al. [21] implemented genetic algorithm where the best match particle pairs are determined by using a fitness function minimizing the total sum of the squares of the particle displacements. This type of fitness function is 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, the algorithm often becomes unsuccessful when the particle displacement and the particle interval come close to each other. So the optimization should be based on some other additional criteria than just the least sum of the distance of the particle pairs. In this aspect, the technique based on Hopfield neural network [22] seems more promising [23]–[25]. The physical constraints applied there accommodate smoothness and rigidity of the displacement fields accompanied by the unique matching partner and the least sum of the distance of the matched pairs. It outperforms the conventional tracking methods but the computational process is so time consuming that very few particles per frame (<150) can be dealt within tolerable computation time [24], [25]. The process becomes tremendously slow for higher number of particles per frame. In order to overcome this problem a separate computational framework based on cellular neural network is proposed here. The proposed algorithm is described in

Sect. 2 and performance test with different particle images are demonstrated in Sect. 3. The work is briefly concluded in Sect. 4.

2. Proposed Algorithm

First of all, the particles in the two frames are mapped as shown in Fig. 1. Numerical figures in the horizontal and vertical directions show the identity numbers of the particles in the first frame and the second frame respectively. All the “possible match vectors” are represented by the small circles. The term “possible match vectors” refers the displacement vectors drawn from a particle in first frame to the particle in second frame. For each particle in one frame, all the particles in another frame are considered as the probable match, and the displacement vectors corresponding to these matches are the “possible match vectors”. For example, in Fig. 1, the first column of the matrix of the circles represents the so-called “possible match vectors” of the particle in the first frame with particle identity number “1”. In another way, the bottommost row of the matrix of the circles represents the “possible match vectors” of the particle in second frame with particle identity number “1”. Each of these vectors is then considered as the neuron unit of the neural network. The neural computational paradigm proposed here is based on cellular neural network (CNN) [26]. In this network, any unit is connected only to its neighbor units, i.e. only the adjacent units interact directly with each other. Units not in the immediate neighborhood have indirect effect because of the propagation effects of the dynamics in the network. Mathematically, if the unit located in position (i, k) of a two dimensional $M * N$ array is denoted by C_{ik} , then its neighborhood N_{ik}^r is defined by:

$$N_{ik}^r = \{C_{jl} | \max\{|j - i|, |l - k|\} \leq r, 1 \leq j \leq M; 1 \leq l \leq N\} \quad (1)$$

where, r denotes the size of neighborhood and is a positive integer number. M and N here represent the number of particles in the first and the second frame.

The basic principle now is that the Lyapunov energy function of the network system should be minimized as formulated below:

$$E = -\frac{1}{2} \sum_{i,k} \sum_{\substack{j,l \\ C_{jl} \in N_{ik}^r}} T_{ik,jl} \cdot V_{ik} \cdot V_{jl} - \sum_{i,k} I_{ik} \cdot V_{ik} \quad (2)$$

5	○	○	○	○	○
4	○	○	○	○	○
3	○	○	○	○	○
2	○	○	○	○	○
1	○	○	○	○	○
	1	2	3	4	5

Fig. 1 Mapping of the particles.

In Eq. (2), V_{ik} is the neuron state at the crossing of row i and column k (so is V_{jl} in like manner). $T_{ik,jl}$ is the connection weight of the two neuron units ik and jl , and I_{ik} the energy threshold of unit ik . The neuron units undergo series of state updates in order to minimize the function, and updated state of neuron units are not computed in a simple threshold scheme (0 or 1) but according to the following normalized function $V_{ik}^{(new)}$:

$$V_{ik}^{(new)} = \frac{1}{2} \left(1 + \tanh \left(\frac{u_{ik}}{0.5} \right) \right) \quad (3)$$

where,

$$u_{ik} = \sum_{\substack{j,l \\ C_{jl} \in N_{ik}^r}} T_{ik,jl} \cdot V_{jl} + I_{ik} \quad (4)$$

Now, an object function should be defined on the basis of the different physical constraint conditions. [23] introduced four different physical constraint conditions for the object function which were further modified by the present authors [24], [25] and are now re-formulated as below in accordance with the proposed new scheme.

$$\phi_0 = C_0 \sum_{i,k} f_r(i, k) \cdot V_{ik} \quad (5)$$

$$\phi_1 = C_1 \sum_{i,k} f_x(i, k) \cdot V_{ik} + C_2 \sum_{i,k} f_y(i, k) \cdot V_{ik} \quad (6)$$

$$\phi_2 = C_3 \sum_{i,k} \sum_{\substack{j,l \\ j \neq i \\ l \neq k \\ C_{jk} \in N_{ik}^r}} |g(i, j) - g(k, l)| V_{ik} \cdot V_{jl} \quad (7)$$

$$\phi_3 = C_4 \sum_{i,k} \left[\sum_{\substack{j \\ j \neq i \\ C_{jk} \in N_{ik}^r}} V_{ik} \cdot V_{jk} + \sum_{\substack{l \\ l \neq k \\ C_{il} \in N_{ik}^r}} V_{ik} \cdot V_{il} \right] \quad (8)$$

The first condition (5) requires the minimization of the total sum of the distances between all the paired particles, where $f_r(i, k)$ is the Euclid distance between the particles i and k in different frames. This criterion addresses the assumptions made by multi-frame tracking methods [2], [3] and genetic algorithm based particle pairing method [21]. This condition alone is not sufficient for the increased numbers of particles to be tracked correctly as mentioned earlier. Hence, second condition (6) is proposed which asks for the minimal sum of the two components of the particle displacement vectors where $f_x(i, k)$ and $f_y(i, k)$ stand for the x-component and y-component respectively of the displacement vector from particle i to particle k . Presence of this condition resolves the ambiguities which can't be dealt with first condition alone.

Additionally, the third condition (7), which asks the minimization of the sum of local fluctuations of the distances to every neighbor particle, ensures the rigidity of the flow pattern. $g(i, j)$ stands for the Euclid distance between

the two particles i and j in the first frame and similarly $g(k, l)$ stands for the Euclid distance between the particles k and l in the second frame, where particle k and particle l are the matched partners of particle i and particle j respectively. This condition addresses the assumptions made in binary-image cross-correlation methods [4] and in relaxation based method [14]–[19] with enough flexibility, and in a global manner. The second and the third conditions are, in some way, involved in the idea of fuzzy logic based tracking [5]–[11]. The application there is rule-based and is quite vague, but here the network understands from the global perspective of the flow.

Finally, the fourth condition (8), requires the uniqueness of solution i.e. one particle in one frame will have only one unique partner in the other frame. So, every row or column of the matrix of the circles in Fig. 1 ought to have at least but no more than one active neuron as shown in Fig. 2. The five coefficients C_0, C_1, C_2, C_3 and C_4 in the abovementioned equations are weight constants by which the weight balance of the four different constraint conditions can be controlled. Using the combination of constraint conditions, the final object function is given by:

$$\phi = \phi_0 + \phi_1 + \phi_2 + \phi_3 \quad (9)$$

Now, by comparing the second-order terms of Eq. (9) with that of the Lyapunov function in Eq. (2), the connection weight is mathematically determined as below:

$$T_{ik,jl} = -\frac{C_3}{2}(1 - \delta_{ij})(1 - \delta_{kl})|g(i, j) - g(k, l)| - \frac{C_4}{2}\delta_{ij}(1 - \delta_{kl}) - \frac{C_4}{2}\delta_{kl}(1 - \delta_{ij}) \quad (10)$$

where, δ_{ij} stands for the Kronecker delta and is equal to 1 if $i = j$ or 0 if otherwise.

Similarly, by comparing the first-order term, the threshold I_{ik} is computed as follows:

$$I_{ik} = -f_r(i, k)C_0 - f_x(i, k)C_1 - f_y(i, k)C_2 \quad (11)$$

By substituting $T_{ik,jl}$ and I_{ik} in Eq. (4) from Eqs. (10) and (11), the unit output u_{ik} in Eq. (4) is obtained. From the unit output u_{ik} , thus obtained, the updated unit value $V_{ik}^{(new)}$ in Eq. (3) is calculated. The series of computation is iterated until the object function ϕ tends to be minimum and the

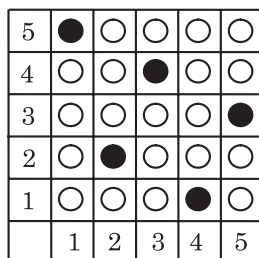


Fig. 2 Network after optimization.

unit values come to their respective optima, which are basically “active” or “inactive”. The initial values of V_{ik} and V_{jl} are given random decimal numbers ranging between 0.0 and 1.0. In order to facilitate the determination of the weight constants, the distance parameters in the object functions are also normalized in the range between 0.0 and 1.0.

The terminal condition of the iterative calculation is that all the “active” neurons remain unvaried over successive iterations. Figure 2 depicts the state of the network after successful optimum solution. In the figure, the black-solid circles represent the “active” neurons and hollow circles represent the “inactive” neurons. Since the unit values in Eq. (3) are not given in binary digits, the maximal unit value for each row (or column) in the abovementioned matrix is regarded as “active” and all the others as “inactive”. To interpret Fig. 2 in terms of particle pairing, black-solid circles represent the most likely match pairs. Figure 2 is the instance of the solution where particles are matched as 1-5, 2-2, 3-4, 4-1 and 5-3, i.e. particle with identity number 1 in first frame is the partner of the particle with identity number 5 in the second frame, and likewise the others.

3. Experimental Results

The proposed algorithm is tested by using the PIV standard images available from the Visualization Society of Japan [27]–[29]. These images are composed of various sets of synthetic time-series particle images generated from Direct Numerical Simulation (DNS) results of an impinging jet in a square cavity. The images come with text files of the original particle coordinates so that one can compare the analysis results of particle matching with the correct data set. Table 1 shows the particle tracking results for the images with different numbers of particles using different algorithms. These are the results for the images from #301 series of the standard image library. Since the objective here is to develop a PTV measurement system without a priori knowledge of the flow field and on the basis of cost function optimization, the present algorithm is compared with the other algorithms having similar strategies. In the table, the abbreviation NALG stands for the new algorithm proposed here, HALG, for Hopfield neural network based algorithm [23]–[25], and GALG, for the genetic algorithm

Table 1 Particle tracking results for different algorithms.

Number of Particles	Algorithm	CPU Time	Error
50	NALG	0"	0
	HALG	9"	0
	GALG	5"	0
100	NALG	2"	0
	HALG	91"	0
	GALG	24"	0
150	NALG	5"	0
	HALG	620"	0
	GALG	33"	10

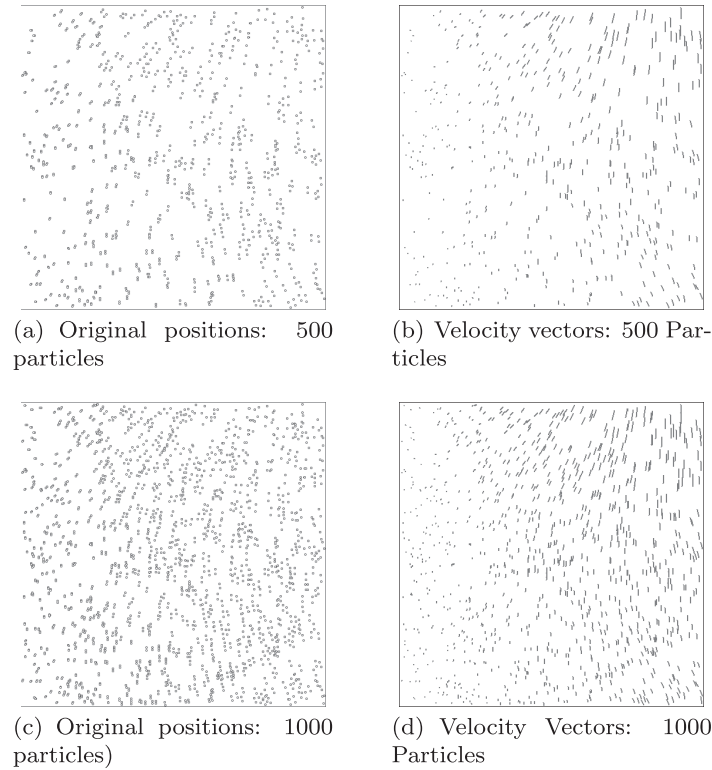


Fig. 3 Performance test with #301 image series.

based technique [21]. Similarly, while mapping the particles as shown in Fig. 1, the identity number of the particles present in the images can be assigned randomly. Our several trials with the random assignment of the identity number of the particles didn't make any change in the performance of the algorithm.

As shown in Table 1, the new algorithm has been successful to track the particles with fine accuracy and encouraging computation time. The computation times shown here are for 1.2 GHz Celeron machine. The algorithm outperforms the Hopfield neural network based particle tracking algorithm (HALG) in terms of computation time. The computation time for 150 particles was 10 minutes 20 seconds using HALG, and for the further increase in number of particles, for example 300, the process goes too irritating with computation time of more than one hour which is just a few second's processing time for NALG. Similarly, if compared to the genetic algorithm based technique (GALG) the proposed algorithm is better in terms of accuracy, computation time as well as reliability. Though GALG goes fast for lower number of particles if compared to HALG but the computation time is still more in comparison to NALG and pairing results are less accurate. The major drawback with GALG is that the results are not reproducible as they depend upon the random numbers generated during the execution of the programs. In a performance test, carried out for the image with 300 particles using GALG, the error in pairing results varied from 12% to 30%. The same performance test for NALG produced the results without any error.

Table 2 Performance test for higher number of particles.

Number of Particles	CPU Time	Error
300	9"	0
500	30"	0
700	45"	0
1000	80"	0

The proposed algorithm is tested for further higher number of particles too as shown in Table 2. The computation time for 1000 particles is just 80 seconds without any pairing errors. The displacement vector maps obtained using NALG are shown in Fig. 3 together with the positions of the particles in the images. The positions of the particles taken at $t = t_0$ and $t = t_1$ are superimposed in a single frame as shown in the figures which in some way reveal the general pattern of the flow. Figure 4 shows the comparisons between HALG, GALG and NALG in terms of computation time.

Additionally, a performance evaluation of each of the sub-function of the proposed objective function has been carried out. Table 3 depicts the scenario when each of the ϕ_1 , ϕ_2 , ϕ_3 and ϕ_4 are removed, one at a time, from the objective function. It can be observed from the table that the importance of the each of the element of the objective function is more pronounced with the increased number of particles in the images. It supports the argument behind the inefficiency of GALG to higher number of particles. It should also be noted here that the function showing the uniqueness has significant influential role in the optimization procedure. This

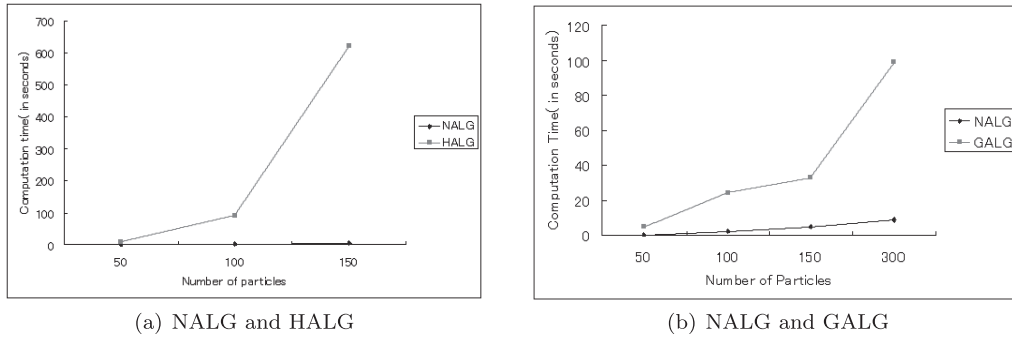


Fig. 4 Comparisons of computation time for NALG, HALG and GALG.

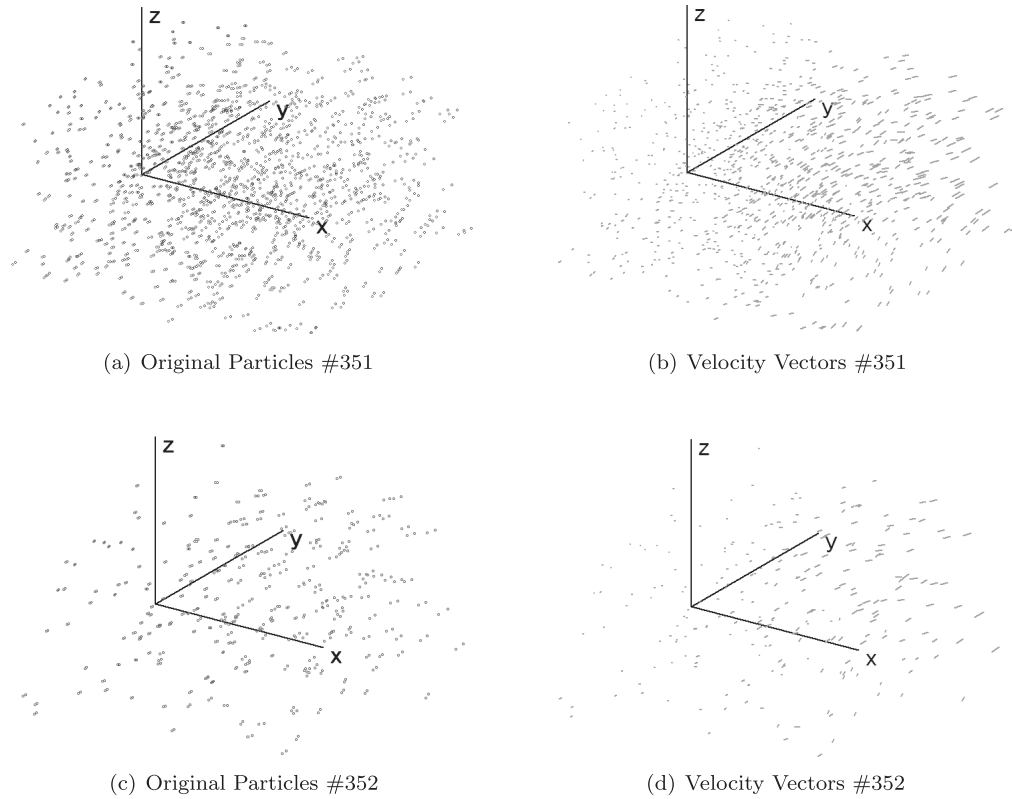


Fig. 5 Performance test with three-dimensional image data.

Table 3 Performance test of the objective function.

Number of Particles	ϕ_1	ϕ_2	ϕ_3	ϕ_4
100	0	2	0	3
300	6	14	1	23
500	13	13	5	47
1000	12	164	13	142

condition of uniqueness is not addressed by many existing algorithms mentioned above including GALG. It is to add here that the weight parameters C_0 to C_4 play an important role in the proposed scheme. This kind of parameter dependency is common to almost every PTV algorithms mentioned in Sect. 1 including GALG and HALG. Normally, the selection of such kind of parameters in PIV is done on

hit and trial basis till the maximum numbers of true vectors are recovered. In the standard images tested here it is easy to verify the true vectors as their correct answer is known while in the case of real-world images it should be done with the aid of some validation algorithms [30], [31].

Similarly, the algorithm is tested with 3-D images too. To test for the 3-D images additional z component is incorporated in distance parameters in Eqs. (5) and (7) and the physical condition expressed in (6) should contain additional $f_z(i, k)$ component like other two components present there. Standard image series #351 and #352 are taken from the standard images library. Figure 5 shows the displacement vectors maps for 250 particles of #352 image series and 1000 particles of #351 image series. The number of errors obtained is actually zero in the case of 3-D images

too. The 3-D coordinates of particles used for the matching operation are obtained from the numerical database of the standard images. But, it should be noted that, in the fully 3-D system the accuracy of the matching is dependant upon the accuracy of the algorithms to extract the three dimensional coordinates of the particles [32]–[34]. But in any case, the constraint conditions proposed here are, obviously, more favourable to three-dimensional flows as particle positions in this case are resolved in depth directions also and hence provide the more accurate estimates of distance parameters.

Finally, as the authors have done experiments with standard images available to check the validity of the PTV algorithms, it should be noted that in real world scenario there is a possibility of some overlapping particles and definitely some such particles in one frame which actually don't have any correct partner in the second frame. Hence in such case of overlapping and missing particles the ambiguous situation may arise producing the outliers. Such kind of outliers should be removed using proper outlier detection algorithms [30], [31], [35]. Similarly, it should also be noted that some optical conditions of recording and computation of the image centroid from such images may also affect the final pairing results. Hence, care should be taken on proper hardware setup and image pre-processing.

4. Conclusions

A new algorithm for particle tracking velocimetry has been proposed. The algorithm doesn't require a priori knowledge of the fluid flow (like maximum flow velocity) as required by many other algorithms and it is accurate to low density as well as high density particle images. The main objective here is to enhance particle matching problem from the view point of cost function optimization. The significant improvement in computation time has cited a comfortable way for the use of this combinatorial optimization approach which was previously considered tedious due to long computation time as evident in the results of Hopfield neural network based PTV. The results are quite reliable and more constraints are effectively being addressed then in genetic algorithm based methodology. Obviously, the constraint conditions proposed here do not address all the domains of applications where PIV system is getting popular. Slight modifications or improvements can be made in the part of the physical constraint conditions to more accurately address the desire of one's system.

References

- [1] M. Raffel, C. Willert, and J. Kompenhans, *Particle Image Velocimetry*, Springer, 1998.
- [2] Y.A. Hassan and R. Canaan, "Full field bubbly flow velocity measurement using a multiframe particle tracking technique," *Experiments in Fluids*, vol.12, pp.49–60, 1991.
- [3] T. Kobayashi, T. Saga, and S. Segawa, "Multipoint velocimetry measurement for unsteady flow field by digital image processing," in *Flow Visualization*, ed. R. Reznicek, pp.197–202, Hemisphere, New York, 1989.
- [4] T. Uemura, F. Yamamoto, and K. Ohmi, "High speed algorithm of image analysis for real time measurement of two-dimensional velocity distribution," in *Flow Visualization*, ed. B. Khalighi, M. Barun, and C. Freitas, pp.129–134, ASME FED-85, 1989.
- [5] M.P. Wernet, "Fuzzy logic particle tracking velocimetry," *Proc. SPIE*, vol.2005, pp.701–708, 1993.
- [6] M.P. Wernet, "Fuzzy logic enhanced digital PIV processing software," in *Technical Memorandum*, ed. N.G.R. Center, National Aeronautics and Space Administration Washington(NASA), United States, 1999.
- [7] M.P. Wernet, "Fuzzy inference enhanced information recovery from digital PIV using cross-correlation combined with particle tracking," *Proc. SPIE*, vol.2546, pp.54–64, 1995.
- [8] M.P. Wernet, "New insights into particle image velocimetry data using fuzzy-logic-based correlation/particle tracking processing," *Experiments in Fluids*, vol.30, pp.434–447, 2001.
- [9] L. Shen, X. Song, and F. Yamamoto, "Particle tracking velocimetry with fuzzy logic," *Proc. ASME/JSME Joint Fluid Engineering Conference*, San Fransisco, California, 1999.
- [10] K. Ohmi and A. Sapkota, "Fuzzy logic algorithm for particle tracking velocimetry," *J. Visualization Society of Japan*, vol.27, no.S-1, pp.167–168, 2007.
- [11] A. Sapkota and K. Ohmi, "Fuzzy logic based particle tracking velocimetry," *Proc. Second International Symposium on Advanced Fluid/Solid Science and Technology in Experimental Mechanics*, Osaka, Japan, 2007.
- [12] M. Ishikawa, F. Yamamoto, Y. Murai, M. Iguchi, and A. Wada, "A novel PIV algorithm with velocity gradient tensor," *Proc. PIV-Fukui*, pp.51–56, Japan, 1997.
- [13] K. Okamoto, W. Schmidl, and Y.A. Hassan, "Least force technique for the particle tracking algorithm," in *Flow Visualization VIII*, ed. J.P. Crowder, pp.647–652, Begell House, New York, 1995.
- [14] S.J. Baek and S.J. Lee, "A new two-frame particle tracking algorithm using match probability," *Experiment in Fluids*, vol.22, pp.23–32, 1996.
- [15] S.J. Baek and S.J. Lee, "Two-frame PIV and its application to a turbulent channel flow," *Proc. PIV-Fukui*, pp.217–226, Japan, 1995.
- [16] H.B. Kim and S.J. Lee, "Performance improvement of two-frame particle tracking velocimetry using a hybrid adaptive scheme," *Measurement Science and Technology*, vol.13, pp.573–582, 2002.
- [17] K. Ohmi and D.H. Lam, "New particle tracking PIV using an improved relaxation method," *Proc. 8th International Symposium on Flow Visualization*, Sorrento, Italy, 1998.
- [18] K. Ohmi and H. Li, "Particle-tracking velocimetry with new algorithms," *Measurement Science and Technology*, vol.11, pp.603–616, 2002.
- [19] K. Ohmi and H. Li, "Particle tracking velocimetry using a discrete relaxation method," *Proc. Sixth Symposium on Fluid Control, Measurement and Visualization*, Sherbrooke, Canada, 2000.
- [20] S.T. Barnard and W.B. Thomson, "Disparity analysis of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.PAMI-2, no.4, pp.333–340, 1980.
- [21] K. Ohyama, T. Takagi, T. Tsukuiji, S. Nakanishi, and K. Kaneko, "Particle tracking technique and velocity measurement of the visualized flow fields by means of genetic algorithm," *J. Visualization Society of Japan*, vol.13, no.S-1, pp.35–38, 1993.
- [22] J.J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat'l Academy of Science*, vol.97, pp.2554–2558, USA, 1982.
- [23] M. Knaak, C. Rothlubbers, and R. Orglmeister, "A hopfield neural network for flow field computation based on particle image velocimetry/particle tracking velocimetry image sequences," *Proc. IEEE International Conference on Neural Networks*, pp.48–52, 1997.
- [24] K. Ohmi, "Hopfield neural network based particle tracking velocimetry," *J. Visualization Society of Japan*, vol.22, no.S-1, pp.209–212, 2002.

- [25] K. Ohmi and A. Sapkota, "Hopfield neural network based particle tracking velocimetry," *Proc. Fifteenth International Symposium on Transport Phenomena*, Bangkok, Thailand, 2004.
- [26] L. Chua and L. Yang, "Cellular neural networks: Theory and applications," *IEEE Trans. Circuits Syst.*, vol.35, no.10, pp.1257–1290, 1988.
- [27] K. Okamoto, S. Nishio, T. Saga, and T. Kobayashi, "Standard images for particle-image velocimetry," *Measurement Science and Technology*, vol.11, pp.685–691, 2000.
- [28] K. Okamoto, S. Nishio, T. Kobayashi, T. Saga, and K. Takehara, "Evaluation of the 3-D PIV standard images (PIV-STD project)," *J. Visualization*, vol.3, pp.115–124, 2000.
- [29] K. Okamoto, S. Nishio, T. Kobayashi, and T. Saga, "Standard images for particle imaging velocimetry," *Proc. PIV-Fukui*, pp.229–236, Japan, 1997.
- [30] A. Sapkota and K. Ohmi, "PTV data validation using fuzzy logic processor," *Proc. Thirteenth International Symposium on Flow Visualization*, Nice, France, 2008.
- [31] J. Nogueira, A. Lecuona, and P. Rodriguez, "Data validation, false vectors correction and derived magnitudes calculation on PIV data," *Measurement Science and Technology*, vol.8, pp.1493–1501, 1994.
- [32] K. Nishino, N. Kasagi, and M. Hirata, "Three-dimensional particle tracking velocimetry based on automated digital image processing," *Trans. ASME, J. Fluids Engineering*, vol.111, pp.384–391, 1989.
- [33] Y. Sata, K. Nishino, and N. Kasagi, "Whole field measurement of turbulent flows using a three-dimensional particle tracking velocimeter," *Flow Visualization*, vol.V, pp.248–253, 1990.
- [34] H. Mass, A. Gruen, and D. Papantoniou, "Particle tracking velocimetry in three-dimensional flows," *Experiments in Fluids*, vol.15, pp.133–146, 1993.
- [35] J. Westerweel, "Efficient detection of spurious vectors in particle image velocimetry data," *Experiments in Fluids*, vol.16, pp.236–247, 1994.



Kazuo Ohmi received his M.Sc. in mechanical engineering from Osaka University, Japan in 1979. He received his Ph.D. in Energetic from Université de Poitiers, France in 1987. He also received doctoral degree in mechanical engineering from Osaka University in 1991. He worked in Laboratoire de Mécanique des Fluides, Université 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, American Society of Mechanical Engineers, American Physical Society and Visualization Society of Japan etc.



Achyut Sapkota received bachelor degree in electrical engineering from Tribhuvan University, Nepal in 2001 and masters of engineering in information systems from Osaka Sangyo University, Japan in 2006. He is currently pursuing doctoral degree at the graduate school of engineering, Osaka Sangyo University, Japan. He worked as a lecturer at Department of Electrical Engineering, Tribhuvan University, Nepal from 2001 to 2003 and also worked in the laboratory of instrumentation and control systems during

that period. He is now a student member of IEEE, IEEJ, and a member of Visualization Society of Japan. His research interests include image processing, artificial intelligence and their application to flow field measurements.