ELSEVIER

PII: S0954(96)00022-2

Artificial Intelligence in Engineering 11 (1997) 135–141

Printed in Great Britain
0954-1810/97/$17.00

Using ART2 networks to deduce flow velocities

K. Jambunathan, V. N. Fontama, S. L. Hartle & S. Ashforth-Frost

Department of Mechanical Engineering, The Nottingham Trent University, Burton Street, Nottingham, UK, NG1 4BU

(Received 1 December 1995; accepted 29 February 1996)

A novel algorithm for obtaining flow velocity vectors using ART2 networks (based on adaptive resonance theory) is presented. The method involves tracking the movement of groups of seeding particles in a fluid space through the analysis of two successive images. Simulated flows, created artificially by shifting the particles through known distances or rotating through known angles, were used to establish the accuracy of the technique in predicting displacements. Accuracies were quantified by comparison with known displacements and were found to improve with increasing displacement, angle of rotation and size of the sampling window. In addition, the technique has been extended to derive qualitative and quantitative information for a practical case of natural convective flow. © 1977 Elsevier Science Ltd. All rights reserved.

Key words: artificial neural networks (ANNs), adaptive resonance theory (ART), flow visualization, particle image velocimetry (PIV), flow velocity, uniform flow, rotating flow, natural convection.

1 INTRODUCTION

Particle image velocimetry (PIV) is a powerful experimental technique that is often used to visualize full field velocity distribution by seeding the fluid with particles. It is an effective approach for obtaining both qualitative and quantitative information regarding a flow regime. Particle tracking velocimetry (PTV) is a similar technique which also uses the images of seeding particles for analysis. PTV tracks the movement of individual particles while PIV monitors groups of particles. Currently, numerical techniques such as cross-correlation, auto-correlation, the Fourier transform method, orthogonal projection of centroids and the particle pathline method can be used to obtain the velocity of these seeding particles.[1–3] Conventional PTV methods that utilize particle streaks to determine velocities tend to be laborious and require the processing of large amounts of data.[4] Furthermore, when the Delaunay net (a numerical method) is used to obtain the directions of the velocity vectors, error propagation leads to reduced accuracy in measured values.

Artificial neural networks (ANNs) are increasingly being applied to pattern recognition problems and their application has recently been extended to PIV. The Hopfield network has been successfully utilized to suppress errors in the determination of 3-D vectors in flows[5] and in obtaining flow directions using the Delaunay net.[6] Although very efficient, this latter application can only be used where the magnitudes, but not the directions, of velocity vectors are known. A modified version of the Kohonen network has been used to deduce fluid flow velocities.[7] The success of this application demonstrates the capability of unsupervised networks to adapt to different flow situations. Its main shortcoming is that there is a 180° ambiguity in the measured flow direction. The backpropagation algorithm has been used to estimate the size distributions of bubbles in a liquid,[8] and to determine the displacement of a group of particles from the complex phase of their pre-processed images.[9]

The application of neural networks to PIV and PTV has been beneficial for a number of reasons. First their capability to suppress noise has been demonstrated.[5,6,9] They have also been used to eliminate the need for sampling windows and their associated problems.[7] Finally, their application has been shown to eliminate many processing steps thus rendering current approaches less laborious.[9] The aim of the current research is to exploit the benefits of ANNs by developing an improved neural networks based methodology for the quantitative and qualitative visualization of natural convection and, ultimately, the simultaneous measurement of temperature and velocity. Preliminary work using neural networks to evaluate fluctuating temperatures in a heated tube has been undertaken.[10,11]

Using the centroids for particle identification, Teo *et al.*[12] trained fuzzy adaptive resonance theory (ART) networks to pair corresponding particles from two successive images. This work demonstrates the potential of fuzzy ART networks in pattern-matching problems.

Since this technique uses the centroids for identification it can only reliably track a particle if it does not move very far from its initial position. Moreover, the accuracy of such a network would be reduced in flows with high seeding densities since it would be difficult for the network to separate particles that are very close to each other. This paper presents a novel technique that uses the ART2 network to deduce flow velocity vectors by tracking the movement of groups of particles. In order to establish its accuracy, the method has been applied to simulations of uniform and rotating flows. It has also been applied to a practical case of natural convective flow generated by a heated tube submerged in a cold water bath.

2 THE ART2 NETWORK

Developed by Carpenter and Grossberg[13] the ART2 network is based on adaptive resonance theory and carries out stable self-organization of recognition codes for arbitrary sequences of input patterns. The ART2 network consists of two main subsystems: the attentional subsystem and the orienting subsystem. The attentional subsystem comprises the $\mathbf{F}_1$ and $\mathbf{F}_2$ layers while the orienting subsystem consists of the **r** layer. In the ART2 network the $\mathbf{F}_1$ layer consists of six sublayers which collectively perform normalization and contrast enhancement of the input pattern. The orienting subsystem contains the **r** layer that contributes to the matching process. There is also an $\mathbf{F}_0$ layer whose sublayers preprocess the input vector before sending it to the $\mathbf{F}_1$ layer. Figure 1 shows a schematic diagram of the ART2 network in which the sublayers in the $\mathbf{F}_0$ and $\mathbf{F}_1$ layers are deliberately omitted for simplicity and clarity.

There are two sets of weights. The bottom-up weights connect the units in the **p** sublayer (in the $\mathbf{F}_1$ layer) to those in the $\mathbf{F}_2$ layer; b_{ij} denotes the weight of the connection from the ith unit in the **p** sublayer to the jth $\mathbf{F}_2$ unit. The top-down weights, on the other hand, connect the $\mathbf{F}_2$ units to those in the **p** sublayer; z_{ji} denotes the weight from the jth $\mathbf{F}_2$ node to the ith node in the **p** sublayer.

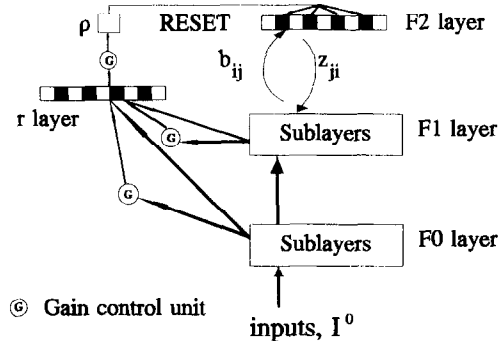

Fig. 1. A simplified schematic diagram of the ART2 network.

The input vector, $\mathbf{I}^0$ is rippled through the $\mathbf{F}_0$ sublayers until the inputs become stable. The output of the $\mathbf{F}_0$ layer is then fed into the **w** sublayer in the $\mathbf{F}_1$ layer. This vector is also rippled through the successive sublayers of the $\mathbf{F}_1$ layer until a stable pattern results. The net effect of the $\mathbf{F}_1$ sublayers is to normalize and contrast enhance **I**, the input vector to the $\mathbf{F}_1$ layer.

The output of the **p** sublayer in the $\mathbf{F}_1$ layer is then propagated to the $\mathbf{F}_2$ layer through the bottom-up weights, b_{ij}. At each $\mathbf{F}_2$ node the net input is given by the weighted sum of the **p** vector and the bottom-up weights, b_{ij}.

$$\text{i.e. } y_j = \sum_{i=1}^{M} p_i b_{ij} \tag{1}$$

where y_j is the net input to the jth $\mathbf{F}_2$ node and M = number of nodes in the **p** sublayer. The $\mathbf{F}_2$ node with the highest sum is chosen as the winner; this winning node is set active while the rest are inhibited. The output of the winning $\mathbf{F}_2$ node, J, is then transmitted back to the **p** sublayer (in the $\mathbf{F}_1$ layer) via the top-down weights, z_{Ji}. At this point the outputs of the **p** sublayer, together with the $\mathbf{F}_1$ input, **I** are propagated to the **r** layer for comparison. If the current $\mathbf{F}_2$ winner is an uncommitted node, or if the current input pattern is similar to that stored in the weights to the current winner, then there will be no reset. The weights to the winning $\mathbf{F}_2$ node are then modified to store the current input vector. The condition for a reset is given by eqn (2):

$$\frac{\rho}{\|r\|} > 1 \tag{2}$$

where ρ is the vigilance parameter. Conversely, if the current vector is dissimilar to the stored pattern the reset condition is satisfied; the system is reset and the current winner is disabled so that it can no longer take part in the competition. The $\mathbf{F}_2$ nodes are then searched until a node with a similar pattern, or an uncommitted node, is found.

The vigilance parameter determines the strictness or coarseness of the classification. If this parameter is high, the classification is very strict and hence only very similar patterns are classified together. On the other hand, if the vigilance is very low then patterns with little similarity will be classified together. After learning, the bottom-up weights to the winning $\mathbf{F}_2$ node are modified as in eqn (3):

$$b_{iJ} = \frac{I_i}{1-d} \tag{3}$$

Similarly, the top-down weights from this winning node are adjusted according to eqn (4):

$$Z_{Ji} = \frac{I_i}{1-d} \tag{4}$$

where d is the output of the winning F_2 node. Further details of the ART2 network can be obtained from Carpenter and Grossberg[13] and Carpenter *et al.*[14]

3 THE NEW TRACKING METHOD

This section presents a novel technique for obtaining flow velocity vectors using ART2 networks. Here, two successive images of a flow are taken and a sampling window is created to capture a group of particles from the first one. The input to the ART2 network is the intensity pattern of this initial sampling window which is stored in the weights after learning. Numerous sampling windows are also chosen from a search region of the second image; the intensity pattern of each window is also fed into the ART2 network. The network compares the pattern of each of these windows with the initial one stored in its weights. If the second pattern is similar to the first then the network selects it as a good match. Otherwise it is rejected.

The search region in the second image can be represented by the set S defined in eqn (5):

$$S = \{(x,y) : x_o - R \leq x < x_o + R; \quad y_o - R \leq y < y_o + R\} \tag{5}$$

where (x_o, y_o) is the position of the initial sampling window from the first image and R is the maximum search radius.

The initial window and those from the search region in the second image are fed to the network. The patterns from the search region which win on the same node as the initial one are selected as the winners. The error (in the pattern space) between each of these winners and the initial sampling window, is calculated. If there is only one winner from the second image then the position of this winning window is considered to be the new location of the particles from the initial window. If there is more than one winner then interpolation is applied in order to determine the best position of the winner, using a parabolic fit. If the distance of the winning position from (x_o, y_o) is $(\Delta x, \Delta y)$ then the velocity will be given by eqn (6).

$$(u,v) = \left(\frac{\Delta x}{\Delta t}, \frac{\Delta y}{\Delta t}\right) \tag{6}$$

where Δt is the time difference between the first and second images.

Figure 2 presents a block diagram of the proposed algorithm to determine the new position of each original sampling window. The intensity pattern of a sampling window taken from the first image is presented to the network. After this pattern is learned the patterns of successive windows, selected from the search region in the second image, are also presented to the network. The ART2 network chooses the patterns that are most

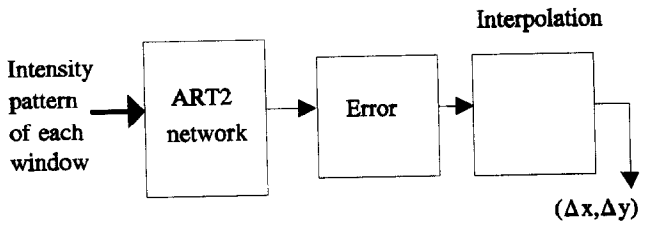

Fig. 2. A schematic diagram of the new algorithm used to derive one velocity vector.

similar to the initial one as the winners. Interpolation is then applied to the coordinates of these winners in order to find the new position of the original particles. This process yields a single velocity vector corresponding to the position of the initial sampling window.

The procedure described above is repeated at different positions in the first image in order to obtain the full field velocity vectors. The complete algorithm is outlined in the flowchart in Fig. 3.

4 DATA PREPARATION

The data used in these experiments were obtained from video images of seeding particles in a flow. Micro-encapsulated liquid crystals, with an average diameter of 100 μm, were used as the seeding particles. Images of the

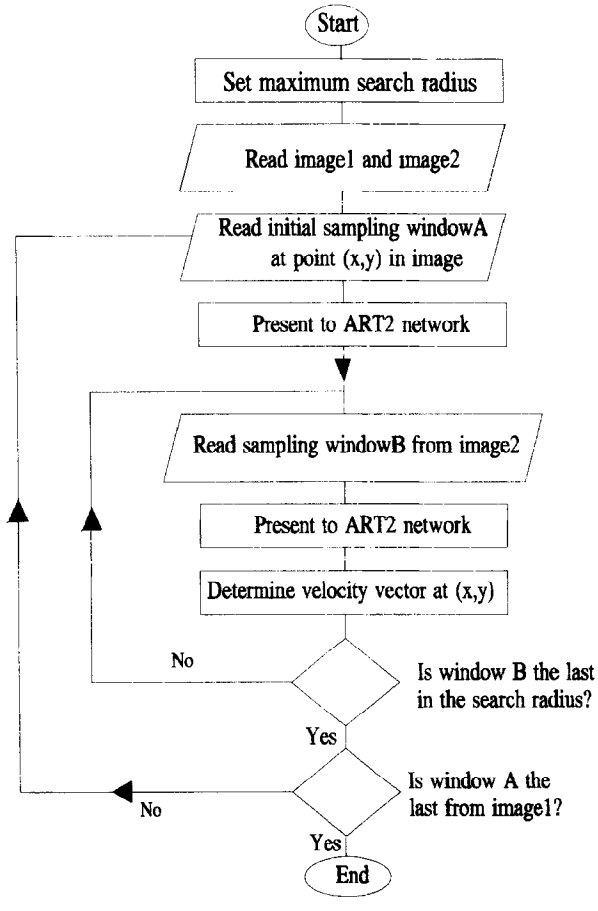

Fig. 3. The complete flowchart of the new algorithm.

flow were taken using a monochrome CCD camera (0·5 Lux sensitivity) and a Sony U-matic video recorder was used to record the images. Two successive frames were then extracted. The time difference between the two images was Δt. More details of the image processing system can be found in previous literature.[15]

Only one image was utilized in simulated flows. A region of this image was manipulated by shifting through known distances or rotation through known angles. After each shift or rotation sampling windows were created in the original and the shifted regions. The size of each sampling window varied between 20×20 and 30×30 pixels. Typically, each sampling window contained the images of approximately three seeding particles. The intensity pattern of each window was represented by a matrix of the same size as that of the sampling window. For instance a sampling window of size 20×20 pixels was represented by a 20×20 matrix. The intensity pattern was measured by the Grey level value at each pixel. Thus, for the ART2 network to learn the pattern of a sampling window, this 20×20 pixel matrix is input to a network with 20×20 nodes in each of its input layers. The value of each of these nodes would be between 0 and 255, representing the Grey level value at each pixel of the sampling window. The number of output nodes determines the number of classes into which the input patterns are to be categorized. In this case only two classes of outputs are necessary. The pattern in the original sampling window is learned and a class is created for it. When each subsequent sampling window is presented the network compares its pattern to the first; if they are similar they are grouped in the same class, otherwise they are rejected and no learning takes place.

5 RESULTS AND DISCUSSION

5.1 Application to simulated flows

Simulated flows, created artificially by shifting the particles through known distances or rotating through known angles, were used to establish the accuracy of the technique in predicting displacements. The network was tested to predict the displacement of these particles and the accuracy quantified by comparing the network's predictions with the known displacements. Two main types of flow were simulated, namely uniform and rotating flows.

5.1.1 Uniform flows

Uniform flow was simulated to quantify the accuracy of the proposed method in tracking particles when they undergo translational motion. The translational motion of particles was simulated by shifting a region of the image through a known distance. This shifted region was placed in the second image. The new method outlined in section 3 was then applied to the original image and its shifted partner in order to determine the displacement. For instance the selected region could be shifted through a distance $(\Delta x, \Delta y)$ and the network tested to determine the exact displacement. The accuracy of this method was calculated by comparing the actual shifts with the values predicted by the network.

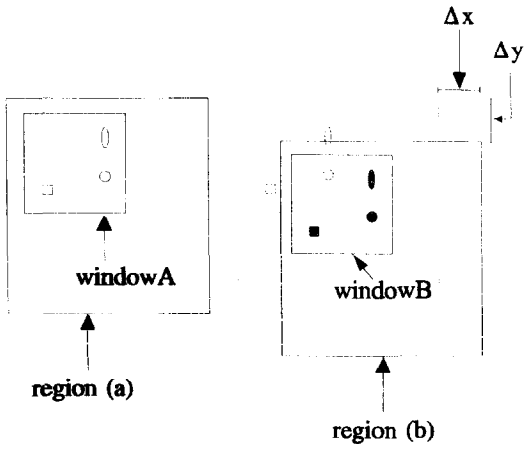

Fig. 4. Illustrating the simulation of uniform flows.

Figure 4 illustrates this simulation. Region (a) from the first image, is shifted through $(\Delta x, \Delta y)$ and placed as region (b) in the second image. A sampling window, window A, is created to capture a few particles in region (a). The intensity pattern of window A is presented to the ART2 network which stores it in its weights after learning. A second window, window B, is created from region (b). Its intensity pattern is also presented to the ART2 network for comparison. If the latter pattern is not similar to that of window A, it is rejected. Otherwise it is retained and the displacements in the x and y directions are obtained from the position of window B.

This method faithfully tracks the movements of particles in these simulations. Its accuracy is seen to increase with increasing displacement of the particles. Figure 5 shows a graph of the average errors obtained from this method when particles are shifted between 2 and 12 pixels. It can be seen that the average error stays very low, varying from 0·61% to 0·104% for displacements between 2 and 12 pixels respectively. The high

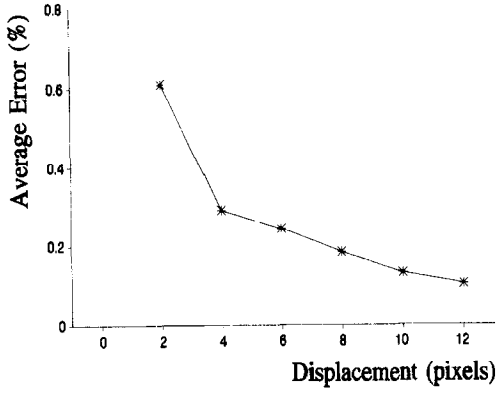

Fig. 5. Illustrating the change in average error with increasing displacements for translational motion.

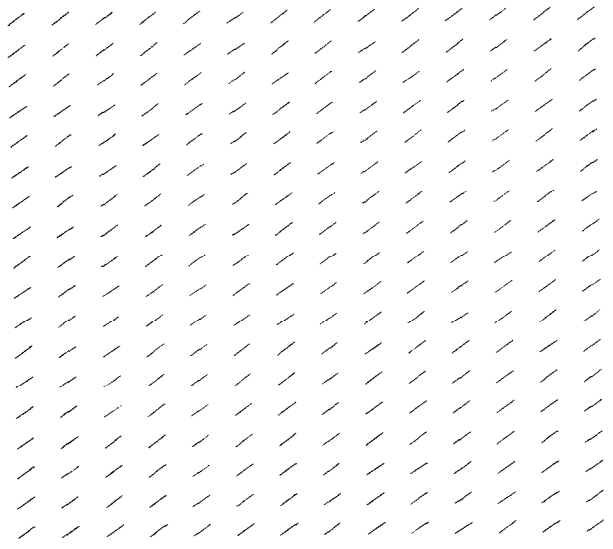

Fig. 6. A typical example of velocity vectors obtained from a simulation of uniform flows.

reliability of this method in tracking particles in translational motion is confirmed by the diagram in Fig. 6 which illustrates typical velocity vectors obtained by this method. In this example the particles were shifted through (4,-4) pixels.

5.1.2 Rotating flows

The new algorithm presented in Section 3 was also tested on a simulation of rotating flows in order to determine its accuracy in tracking particles in such flows. Simulations of these flows were achieved by rotating a region of the image (together with its seeding particles) through known angles.

The simulation of these flows is represented in Fig. 7. Region (a) is rotated through a known angle and then placed as region (b) in the second image. After the ART2 network learns the intensity pattern of window A, sampling windows from region (b) are then inspected until window B, which corresponds to the rotated pattern from region (a), is located. Knowing the displacement of window B from window A, the angle of rotation predicted by the network can be determined.

The accuracy of this method generally increases with

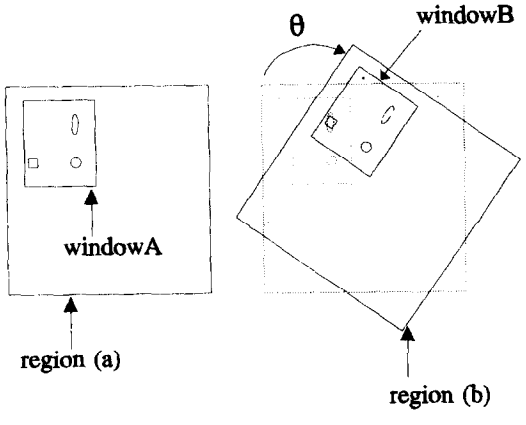

Fig. 7. Demonstrating the simulation of rotating flows.

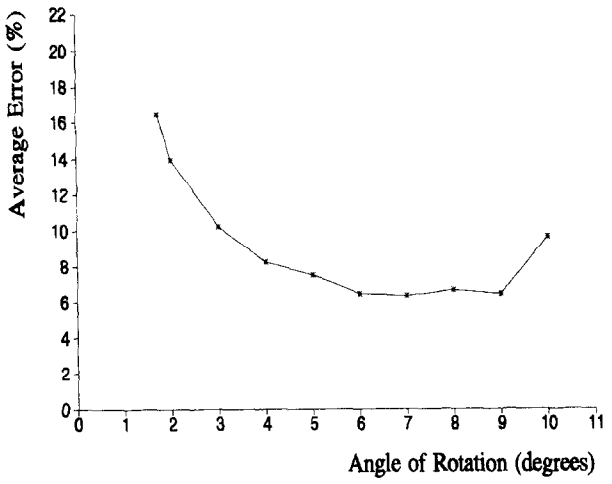

Fig. 8. The variation of average error with the angle of rotation for simulations of rotating flows.

increasing angles of rotation. However, after 7° the accuracy gradually decreases with increasing angles of rotation. Figure 8 illustrates the change in average error with angle of rotation. It can be seen from this diagram that the average error decreases from 16·4% to 6·3% for rotations between 1·72° and 7° respectively. After 7° of rotation the error gradually increases to 9·6% at an angle of 10°. Figure 9 also shows the velocity vectors obtained after rotation using this new method. The plot in this figure demonstrates the vectors obtained when a region of the original image was rotated through 8°.

5.2 The effect of the size of the sampling window

The size of the sampling window influences accuracy of the current technique. The effect of its dimensions on the cross-correlation method has been studied by Kimura and Takamori.[16] Their work demonstrated that the accuracy of the cross-correlation technique decreases with decreasing size of the sampling windows. This is undesirable for two reasons. First, the velocity obtained at each point represents the mean velocity of the particles in the sampling window. Thus, for larger

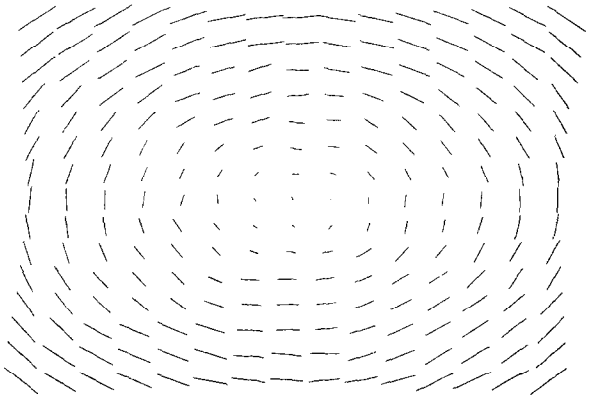

Fig. 9. A typical example of vectors obtained from a simulation of rotating flows.

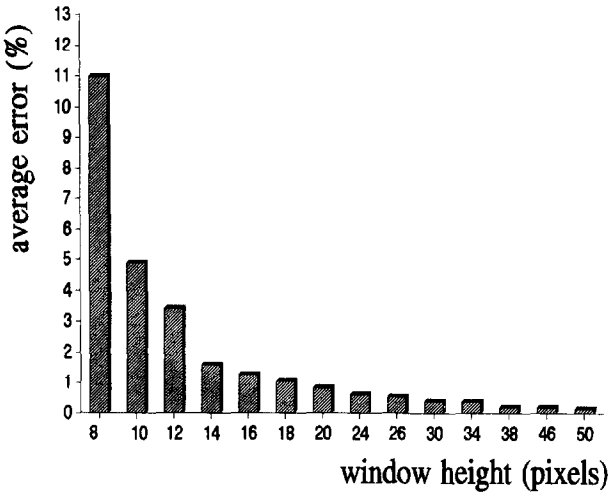

Fig. 10. The effect of the size of the sampling window on the accuracy.

windows the resolution of the measurements is lower. Smaller windows would lead to higher resolution of the measurements. Secondly, smaller windows also lead to fewer inputs, and hence, shorter processing times. Therefore it would be advantageous if the accuracy still remained high at smaller window sizes. Against this background the effect of the size of the sampling window on the accuracy of the new method has been investigated. Windows of sizes ranging from 8×8 to 50×50 pixels were studied. For each window size, a group of particles was captured in the window and the new method (developed in this work) was applied. The average error was calculated for velocity vectors taken at many different points of the initial image. This procedure was repeated for windows of different sizes.

The accuracy of the current method has been found to increase with an increase in the size of the sampling window. Figure 10 shows the average error decreasing from 10·9% to 0·142% for window sizes of 8×8 to 50×50 respectively.

5.3 Application to natural convection

The new technique reported in this work has also been applied to a real natural convective flow. This section describes the use of the method to derive qualitative and quantitative information from this flow.

5.3.1 The experimental rig

The natural convective flow was generated by a uniformly heated and steel cylinder submerged in a cold water bath[15] and is illustrated in Fig. 11. The test cell, which was made of acrylic, measured $150 \times 186 \times 100$ mm and contained cold water at approximately 25°. Hot water, at approximately 50°C, was pumped through the submerged cylinder. The water in the test cell was seeded with 0·01% temperature sensitive microencapsulated liquid crystal particles

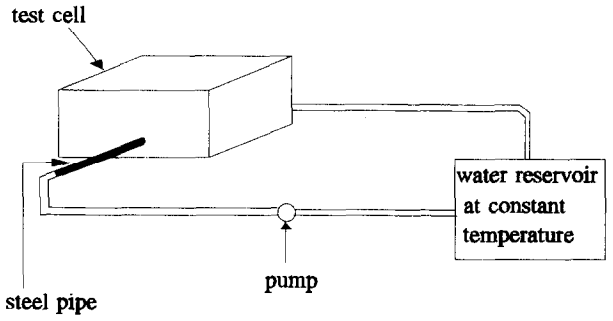

Fig. 11. The experimental rig used to generate natural convective flows.

(measuring on average 100 μm in diameter). A white light source was used to illuminate the flow and a monochrome CCD camera (0·5 Lux sensitivity) to film the flow pattern. Two successive frames were then captured from the video using a Data Translation DT2871 image board. The time difference between the two frames was 0·5 s. These two frames, each with a resolution $512 \times 512 \times 8$ bits were stored on a DEC3000 workstation for analysis. Figure 12 illustrates a typical image of the flow investigated.

5.3.2 Results

In this experiment the heated cylinder submerged in the cold water bath, creates a flow due to natural convection. The fluid density in the immediate vicinity of the hot cylinder is reduced and hence particles rise to the top. The surfaces of the test cell combine to create two areas of recirculation near the top of the water level, one on each side of the cylinder. Figure 13 displays the flow pattern generated by the new method using the ART2 network. This present method displays the two regions of recirculation clearly, confirming its capability to follow the movement of particles even in regions of rapidly changing velocities.

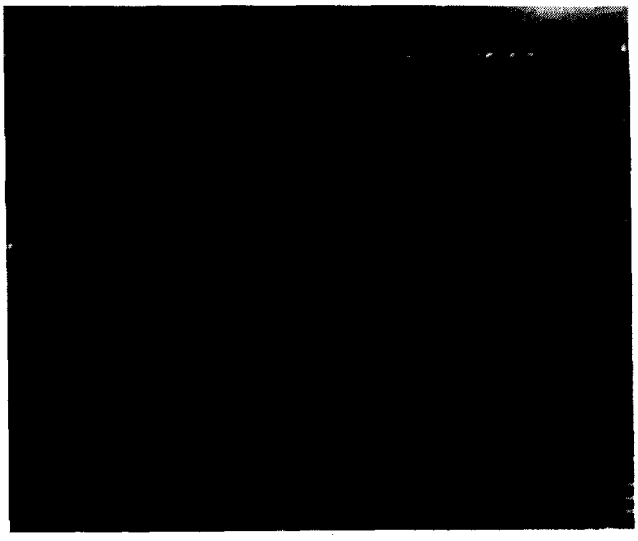

Fig. 12. A typical image of the flow used for this analysis.

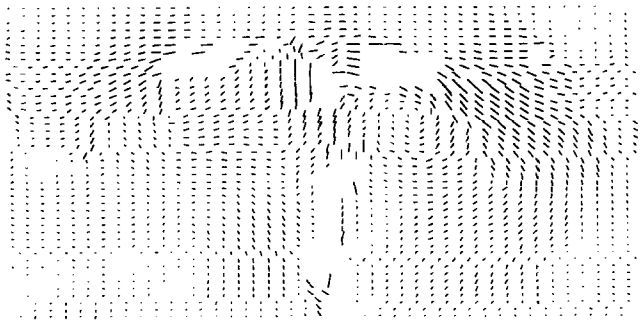

Fig. 13. Vectors of a true natural convective flow obtained using 30 × 30 pixel sampling windows.

6 CONCLUSION

A new algorithm for obtaining flow velocity vectors using ART2 networks has been proposed. This method has been successfully applied to simulations of uniform and rotating flows in order to establish its accuracy in tracking the movement of seeding particles. For simulations of uniform flow the average error decreases from 0·61% to 0·104% for displacements between 2 and 12 pixels respectively. In rotating flows the average error varies from 16·4% to 9·6% for rotations between 1·72° and 10° respectively. Furthermore, the accuracy increases with the size of the sampling window, varying from 89·1% to 99·85% for window sizes from 8 × 8 to 50 × 50 respectively. Finally this technique has been successfully employed to derive qualitative and quantitative information from a real natural convective flow.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Higher Education Funding Council for financial support towards this project, and Mr X. Y. Ju for his invaluable assistance.

REFERENCES

1. Keane, R. D. & Adrian, R. J., Theory of cross-correlation analysis of PIV images. In *Flow Visualization and Image Analysis*, ed. F. T. M. Nieuwstadt. Kluwer Academic, London, 1993, pp. 1–25.
2. Cenedese, A. & Paglialunga, A., Digital direct analysis of a multiexposed photograph in PIV. *Experiments in Fluids*, 1990, **8**(5), 273–80.
3. Murata, S., Study of unsteady separated flows by numerical analysis and image processing. PhD dissertation, Kyoto University, Kyoto, 1993.
4. Laurenco, L., Recent advances in LSV, PIV and PTV. In *Flow Visualization and Image Analysis*, ed. F. T. M. Nieuwstadt. Kluwer Academic, London, 1993, pp. 81–100.
5. Kimura, I., Bandoh, R. & Kuroe, Y., Measurement of 3-D velocity vectors based on spatio-temporal image correlation: processing of erroneous vectors using neural network. *Experimental and Numerical Flow Visualization*, 1993, **172**, 121–6.
6. Murata, S., Kise, H. & Miyake, H., Automatic method for determining flow directions by using a neural network model. In *Proc. The 3rd Triennial International Symposium on Fluid Control, Measurement and Visualization*, San Francisco, 1991, pp. 657–62.
7. Grant, I. & Pan, X., The neural network method applied to particle image velocimetry. In *Proc. SPIE Int. Conf. Optical Diagnostics in Fluid & Thermal Flow*, San Diego, 1993, SPIE 2005, pp. 437–47.
8. Kanitkar, U. & Dudgeon, J., A Walsh transform neural network method for estimating the size distribution of bubbles in a liquid. In *Proc. IEEE SOUTHEASTCON '92*, Birmingham, Alabama, 1992, Vol. 1, pp. 20–2.
9. Yamamoto, T. & Murata, S., The method of velocity measurement with learning function by neural network. *Journal of Visualization of Japan*, 1993, **13**(2), 103–6.
10. Jambunathan, K., Hartle, S. L., Ashforth-Frost, S. & Fontama, V. N., Evaluating convective heat transfer coefficients using neural networks, *International Journal of Heat and Mass Transfer* (in press).
11. Ashforth-Frost, S., Fontama, V. N., Jambunathan, K. & Hartle, S. L., The role of neural networks in fluid mechanics and heat transfer. In *Proc. IEEE Instrumentation and Measurement Technology Conference*, Waltham, MA, 23–26 April, 1995, Ch. 157, pp. 6–9.
12. Teo, C. L., Lim, K. B., Hong, G. S. & Yeo, M. H. T., A neural net approach in analyzing photographs in PIV. *International Conference on Systems, Man, and Cybernetics*, Charlottesville, VA, 1991, Vol. 3, pp. 1535–8.
13. Carpenter, G. A. & Grossberg, S., ART2: Self-organization of stable category recognition codes for analogue input patterns. *Applied Optics*, **26**, 4919–30.
14. Carpenter, G. A., Grossberg, S. & Rosen, D. B., ART2-A: an adaptive resonance algorithm for rapid category learning and recognition. *Neural Networks*, 1991, **4**, 493–504.
15. Ashforth-Frost, S., Dobbins, B. N., Jambunathan, K., Wu, X. P. & Ju, X. Y., A comparison of interrogation methods for particle image velocimetry. In *Proc. SPIE Int. Conf. Optical Diagnostics in Fluid and Thermal Flow*, San Diego, 1993, SPIE 2005, pp. 478–89.
16. Kimura, I. & Takamori, T., Image processing of flow around a circular cylinder by using correlation technique. In *Flow Visualization IV*, 1987, pp. 221–6.