An Image Processing System for Quantitatively Analyzing Two-Dimensional Fluid Velocity-Field Image

Zhongquan Wu, Junhua Zhu, Su Chen, Lin Yang
Department of Information Electronics

Hongqing Xu, Jinlong Yang
Department of Mechanics
Tsinghua University
Beijing, People's Republic of China

ABSTRACT

The quantitative analysis of 2D fluid velocity field plays an important role in the investigation of fluid mechanics and other areas. In this paper a new system for analyzing 2D fluid image is presented. The 2D flow visualization image is input through a TV camera, and then followed by a set of processes to extract the velocity information, finally the principal parameters of the flow field are computed and displayed. In this system the control strategy is mixed bottom-up and top-down and the knowledge of the fluid mechanics is introduced on several levels to improve the performance of the system. For this reason, it is more efficient and has more powerful functions than the conventional ones by reducing the human intervention.

KEYWORDS: image processing, flow visualization, velocity field, line feature extraction, velocity direction determination

1. Introduction

The quantitative analysis of the 2D fluid flow plays an important role both in the investigation of the flow mechanism, and in many application areas, such as machinery, chemical processing, hydraulics, etc. [1]. The hot-wire anemometer and laser Doppler anemometer are advanced instruments for the measurement of dynamic flow velocity. Although the accuracy of them is high enough, only one or a few velocities can be measured at one time. In the case of some complicated fluid flow, such as unsteady flow field, velocity distribution of the whole flow can not be measured by the above instruments instantaneously. In order to study the whole dynamic flow, we must measure the velocities simultaneously. Flow visualization techniques provide useful information about the behavior of the whole flow field, but they are not usually intended to provide quantitative velocity data. Recent developments of image processing techniques has been promoted in the flow measurement. Some researchers applied the speckle method to the measure of the fluid velocity in the 70's and 80's [2]. In the early 80's Kensaku Imaichi, Kazuo Ohmi and et al. developed some new systems for computing some of the principal physical variables of fluid by combining the flow visualization and image processing techniques [3, 4]. These techniques take the advantage of measuring the whole 2D flow field at the same time, whether steady or unsteady. In comparison with the point by point measurement, it is a tremendous progress, but there is a tedious and time consuming input procedure with too much man-made intervention in the above system, because the tracer pathlines are input by the manual operation on the digitizing tablet.

In this paper, a new flow image analysis system for 2D dynamic flow is presented. The main goal of the research described here is to develop a more efficient and flexible system in this domain. There are two important problems to be considered. First, although there are a lot of image processing techniques [6, 7], most of them have to be modified or improved in order to fit the requirements for the quantitative analysis of the flow velocity-field. In some steps, we have to develop some new schems for our system such as the direction determination, etc. Second, it is necessary to represent the domain-specific knowledge in some way, so this knowledge can be used during the analysis to make the system more efficient and robust. The control strategy of this system is mainly bottom-up as shown in the major block diagram (see Fig. 1). The control proceeds from the noisy grey-level image to line structures and finally to the descriptions—the physical parameters of the fluid flow. The data flow is passed from the low level to the higher one under the image driven mode. But in some steps, the image processing procedure is model-directed, i.e., top-down; knowledge of the geometry and the fluid mechanics has directed the analysis to improve the accuracy and reliability here. To sum up, this system involves the control of a large and complex image processing tasks. To achieve a more automatic and powerful system, we use the mixed bottom-up and top-down control strtegies here. In the next sections we will give some details of several important steps of this system.

2. Pathline Enhancement and Feature Image Extraction

It is difficult to extract the pathlines directly because some pathlines in the image are degraded by noise and interference. Therefore some preprocessings must be undertaken before extracting the pathlines.

2.1. Preprocessing

Because of the illumination variation of the light source, the background intensity is often nonhomogeneous. An improved local contrast stretching method is used to remove this nonhomogeneity. The main idea of this method is to use the mean of grey levels in a neighborhood as the estimated background intensity and use the grey level variance of the same neighborhood for estimating the intensity variation caused by the tracer pathlines. Suppose the grey level of the central point of the neighborhood is $f(x,y)$, and the local mean grey level is $f_{av}(x,y)$, the following transformation is implemented [5]

$$g(x,y) = \begin{cases} g_{bk} + K_1 \cdot \text{Max}\left\{0, [f(x,y) - f_{av}(x,y)]\right\}, & \sigma_f > V_{thr} \\ g_{bk} + K_2 \cdot \text{Max}\left\{0, [f(x,y) - f_{av}(x,y)]\right\}, & \text{otherwise} \end{cases}$$

where $g(x,y)$ is the new image grey level, σ_f is the variance of the grey levels in the neighborhood; K_1 and K_2 are gain factors, generally $K_1 > K_2$; g_{bk} is a preset background grey level we assume to take after the preprocessing. As seen in Figure 2 (a), (b), the experimental result is good enough for the further processing.

2.2. The Consistent Relaxation Algorithm

Based on [9] our improved consistent relaxation algorithm is used to enhance the pathlines. The algorithm is an iterative process which consists of the following two procedures.

(1) The initial labeling procedure

A set of templates shown in Figure 3 are designed to detect line structures in the image. These templates take the advantage of the uniform response to the lines widths. Their applicability is excellent. The line strengths in four orientations are denoted by S_k (k = 1,2,3,4) which can be calculated as follows:

a) For 0° and 90° orientation:

$$A_1 = A_{11} + A_{12} + A_{13} \qquad C_1 = C_{11} + C_{12} + C_{13}$$

$$A_2 = A_{21} + A_{22} + A_{23} \qquad (2\text{-}1)$$

$$B = B_1 + B_2 + B_3 \qquad C_2 = C_{21} + C_{22} + C_{23}$$

For 45° and 135° orientation:

$$A_1 = A_{11} + A_{12} + A_{13} \qquad C_1 = C_{11} + C_{12} + C_{13}$$

$$A_2 = A_{23} + A_{22} + 1/2(A_{21} + A_{24}) \qquad (2\text{-}1)$$

$$B = B_1 + B_2 + B_3 \qquad C_2 = C_{23} + C_{22} + 1/2(C_{21} + C_{24})$$

b)

$$S_k = T[B - \min(A_1, A_2),\ B - \min(C_1, C_2)] \quad (k = 1,2,3,4)$$

where the truncating operator T[,] is defined as

$$T[x,y] = \begin{cases} x + y & \text{if } x \geq 0 \text{ and } y \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

A complex image $d(i)$ is formed, where i represents the ith pixel of the image, and suppose the image size is $N \times N$, we have

$$d(i) = \sum_{k=1}^{4} S_k(i) \exp[j\,\pi\,(k-1)/2] \quad 1 < i < N^2 \qquad (2\text{-}2)$$

We can normalize $d(i)$ to obtain an initial iterative image.

$$e^0(i) = d(i) / \max(|d(m)|) \quad 1 < m < N^2 \qquad (2\text{-}3)$$

The magnitude of $e^0(i)$ gives the line strength of the ith point, while the angle of $e^0(i)$ gives the direction in which the tracer pathline, if there is one, passes through that point.

$$\psi = \theta / 2 \qquad (2\text{-}4)$$

where ψ is the actual angle of the pathline ($0° \leq \psi < 180°$) and θ is the phase angle of $e^0(i)$.

(2) The iteration procedure

Starting from e^0, a series of complex images $e^0, e^1, \ldots, e^r, e^{r+1}, \cdots$ are obtained

$$f_k^r(i) = \sum_{j \in N_i} C(k,i,j)\,|e^r(j)| \qquad k = 1,2,3,4 \qquad (2\text{-}5)$$

$$g_k^r(i) = \max\{0, f_k^r(i)\} \qquad k = 1,2,3,4 \qquad (2\text{-}6)$$

$$q^{r+1}(i) = \sum_{k=1} g_k^r(i) \exp[j(k-1)\,\pi/4] \qquad k = 1,2,3,4 \qquad (2\text{-}7)$$

$$e^{r+1}(i) = \begin{cases} q^{r+1}(i) & |q^{r+1}(i)| <= 1 \\ q^{r+1}(i) / |q^{r+1}(i)| & |q^{r+1}(i)| > 1 \end{cases} \qquad (2\text{-}8)$$

where N_i is the 3×3 neighborhood of the point i. The $C(k,i,j)$ in (2-5) is called a consistent factor and $C(k,i,j)$ is given by

$$C(k,i,j) = \begin{cases} w_i \cos(\theta_i - 2\psi_k) & j = i \\ w_j \cos(\psi_{ij} - \psi_k)[\cos(\theta_j - \theta_i) + (\theta_j - 2\psi_k)] & j \in N_i \text{ and } j \neq i \\ 0 & \text{otherwise} \end{cases} \qquad (2\text{-}9)$$

where θ_i, θ_j are the angles of $e^r(i)$, $e^r(j)$ respectively, ψ_k is computed by

$$\psi_k = (k-1)\,\pi/4 \qquad k = 1,2,3,4 \qquad (2\text{-}10)$$

ψ_{ij} is the angle between horizontal direction and the line connecting points i and j. A simple explanation of the meaning of $C(k,i,j)$ is : for the center i of a neighborhood, $C(k,i,j)$ will get its maximum when θ_j is equal to θ_i , and it is coincident with the direction of the line connecting points i and j, as well as the direction to be considered.

Through the iterations, the pathlines contained in the original image are gradually enhanced as shown in Figure 4. The convergence of this process can be shown using the method similar to the one in [10].

2.3. Feature Image Extraction

A binary feature image $B(i,j)$ can be obtained by multiple thresholding the image enhanced by the above consistent relaxation. The advantage of multiple thresholding is in its efficiency in extracting the pathlines while reducing the possible connected pathlines which may appear by using the simple thresholding. Each 8-connected region (a set S) in the

image $B(i,j)$ represents a tracer pathline. First, for each connected set in the feature image, a straight line of the form $y = ax + b$ is estimated by grey-level-weighted least-square method, which can be implemented by minimizing the following evaluation function

$$E = \sum_{(x_i,y_i)\in S} g(x_i,y_i)(y_i - ax_i - b)^2 \qquad (2\text{-}11)$$

Then two end points of this point set are selected based on the criteria that they can not be too far from the line approximated and that their grey-level can not be too low either. Their projections on the estimated line yield the starting and ending points of the pathline.

3. Extracting the Direction Information of the Velocities

To determine the starting point of a pathline, a two step procedure is adopted because of the ambiguity of the velocity direction: first, the velocity direction is initially labeled by introducing the knowledge about the fluid mechanics; then a relaxation process is performed to get a global consistent flow field. We will present two different methods for the first step in the next two subsections.

3.1. The Projection Method

For the image which is taken under a shot flash plus a little bit longer illumination, the pathlines always consist of a bright round point and a trailing line. Usually the diameter of the round point is greater than the width of the line, see Figure 5(a).

The starting point is determined by the help of its grey-level projection onto the principal direction. Obviously, the peak of the projection curve corresponds to the starting point, see Fig. 5(b). An algorithm is developed to detect the location of the peak, i.e., the location of the starting point.

Algorithm 3-1:

1. Suppose a pathline—a feature point set S is found. Compute the mean of grey-levels g_{av} and the standard deviation STDD of this set.
2. Compute the grey-level projection of the set S onto the principal direction, and calculate the summation of the projections within the two end intervals(Δ_0 and Δ_1). Suppose they are s_0 and s_1 respectively.
3. If $| s_0 - s_1 | > X_k STDD$ and $s_0 \geq s_1$, then the starting point is p_0 , otherwise is p_1 . Goto 5.
4. If it is difficult to determine the velocity direction only based on the pathline itself because of the noise, then the velocity direction of the previous adjacent pathline is used to determine the direction of this one.
5. Search the next pathline and determine its direction.

The sizes of Δ_0 and Δ_1 depend on the length of the pathline, usually 3-4 pixels, and the X_k depends on its average width.

An example of this procedure is shown in Fig. 5. Since sometimes the flow field is much more complicated, it is necessary to use the relaxation procedure to adjust the directions of those incorrectly directed pathline remained.(see 3.3)

3.2. The Control Point Directed Method

When the flow image is taken without a shot flash, the above method can not be used for determining the flow direction. Although there are no general solutions for this situation, for certain kinds of flow fields, the knowledge about the fluid flow may be used as an aid to the direction discrimination. Here the high-level model in the knowledge base generates expectations of the flow directions. The process here is the verification of the consistency between the hypothesized directions with those expectaions under the model-directed mode.

First, the directions of velocities at the boundary are determined based on the boundary condition, e.g. basically they are from left to right at the left boundary. Second, a set of control points P_i are preset by giving their coordinates (x_i, y_i), i=1,2,...,N. The corresponding weights of their control ranges W_i and rotation directions (clockwise or counter-clockwise) R_i are specified. Then for each pathline, its weighted distances to all the control points are calculated and compared. Finally, the direction of any pathline is chosen so that it is consistent with the rotation direction of the nearest control point. As a rule of thumb, vortex centers may be chosen as control points. The directions can be determined by the following algorithm.

Algorithm 3-2:

1. For each pathline l_j, compute the coordinates of its midpoint, and the vector from the start to the end v_j.
2. Compare all the weighted distances (weighted by W_i of each control point P_i) to find the least weighted distance and its control point as shown in Fig. 6.
3. After having found the nearest control point, compute the distance from it to the midpoint of the pathline v_j.
4. Calculate the absolute value of the cosine of the angle extented by r_{ij} and v_j, if this value is greater than a threshold, then do (1) otherwise do (2).

(1) In this case, v_j is almost parallel to the r_{ij} based on the principal direction of the flow. If that direction is very close to the principal direction, keep it unchanged, otherwise inverse it.

(2) Compute the rotation direction of the v_j around the control point with the vector r_{ij} directed to its midpoint. If this rotation direction is the same as that of the control point, the velocity direction is correct, otherwise inverse the direction of that velocity.

5. Goto 1, until every pathline has been checked.

The vortex centers can be selected as the control point as shown in Fig. 7. The pathline v_a is close to the up control point and the angle extended by v_a and r_a is close to right angle, so the direction of v_a must be from left to right based on the rotation rule. The pathline v_b is similar to v_a. The pathlines v_c and v_e belong to the region of the principal flow , and the angles between v and r are very small, so their directions can be determined according to the principal direction.

It will be shown in the experiments that the velocity directions can be more reliablely determined under the guidance of the knowledge of the fluid mechanics introduced here.

3.3. The Relaxation Adjustment of the Velocity-Direction

A well designed relaxation algorithm has been developed to adjust those incorrectly directed pathlines after the initial direction determination because the noise interference. This algorithm is based on the fact that the flow directions have to be consistent in any local region similar to the optical flow [10]. There are three kinds of relations between any two pathlines: consistent, unconsistent and independent as shown in Fig 8. The consistent function is defined as follows:

$$C(v_i, v_j) = W(d_{ij})\cos\theta \tag{3-1}$$

where v_i and v_j are two velocity vectors, θ is the angle extended by them. $W(d_{ij})$ is the weight function decreasing as the distance d_{ij} is increasing:

$$W(d_{ij}) = B\exp(-Ad_{ij} / v_i) \tag{3-2}$$

Where A and B are constants, v_i is the length of v_i. The above definition means that the effect which v_j acts on v_i is decreasing as the d_{ij} is increasing. The confidence of the velocity direction can be computed using the following iteration formula

$$C_f^{(r+1)}(v_i) = C_f^{(r)}(v_i) + \underset{j\in N}{C}(v_i, v_j)\, C_f^{(r)}(v_j) \tag{3-3}$$

After comparing the confidence $C_f^{(r+1)}$ with a threshold, if it is lower than the later, the direction of the lines are inversed. A more consistent flow field can be obtained after a few iterations. An experimental result is given in Figure 9.

3.4. Interactive Man-Machine Local Modification

If the quality of the flow image is not good enough, there may be still some inaccurately extracted pathlines, so we have to do some local modifications by man-machine interaction to get a perfect pathline image, as shown in Figure 10.

4. Velocity Field Reconstruction and the Calculation of Principal Physical Variables

The direction of the pathline represents the direction of the velocity while its length is the distance which the tracer has moved in the exposure interval. Thus the actual velocities can be computed by the length of the pathline divided by the exposure interval. Based on those randomly measured tracer velocities, the flow velocities at mesh points can be estimated. Then the stream function and the pressure at mesh points can be computed using the line-integration method.

The following second order linear approximation model is applied to compute the velocities at mesh points.

Suppose $P_o(x_o, y_o)$ is the coordinates of a mesh point, and $P(x_k, y_k)$ is a neighboring point of P_o with the known velocity. Let

$$\begin{aligned} \Delta x_k &= x_k - x_o \\ \Delta y_k &= y_k - y_o \end{aligned} \tag{4-1}$$

Then based on Taylor series expansion we have:

$$\begin{aligned} u(x_k, y_k) &= u(x_o, y_o) + \partial u / \partial x \mid_{p_o} \Delta x_k + \partial u / \partial y \mid_{p_o} \Delta y_k \\ &+ \partial^2 u / \partial x \partial y \mid_{p_o} \Delta x_k \Delta y_k \\ &+ \partial^2 u / \partial x^2 \mid_{p_o} \Delta x_k^2 + \partial^2 u / \partial y^2 \mid_{p_o} \Delta y_k^2 \end{aligned} \tag{4-2}$$

$$\begin{aligned} v(x_k, y_k) &= v(x_o, y_o) + \partial v / \partial x \mid_{p_o} \Delta x_k + \partial v / \partial y \mid_{p_o} \Delta y_k \\ &+ \partial^2 v / \partial x \partial y \mid_{p_o} \Delta x_k \Delta y_k \\ &+ \partial^2 v / \partial x^2 \mid_{p_o} \Delta x_k^2 + \partial^2 v / \partial y^2 \mid_{p_o} \Delta y_k^2 \end{aligned} \tag{4-3}$$

Where $u(x_k, y_k)$, $v(x_k, y_k)$ are the x and y components of the nonzero velocity at some mesh points. They have been obtained in the previous step. Based on the uncompressibility of the fluid flow, the following equation has to be satisfied

$$\frac{\partial u}{\partial x}\Big|_{p_o} + \frac{\partial v}{\partial y}\Big|_{p_o} = 0 \tag{4-4}$$

Solving equations (4-2), (4-3) and (4-4) we may estimate the velocity and its partial derivatives at each mesh point P_o by means of the least-squared method. The estimated velocity distribution at mesh points is shown in Fig. 11. Then the velocity at any point in the 2D flow field can be interpolated from the velocities at the mesh points.

Finally, the stream function and the pressure can be computed by the line integration method, e.g. for the stream function we have

$$\psi = \psi_0 + \int_0^P (udy - vdx) \tag{4-5}$$

where the ψ_0 is the value of stream function at the origin of the flow field, usually the initial value of ψ_0 is supposed to be zero. The origin of the coordinate system is located at the center of the image and the integral path is shown in Fig. 12(a). The stream lines are shown in Figure 12(b). These computational results are very useful for the fluid flow analysis.

5. Conclusion

A new image processing system has been developed for computing the principal physical parameters of 2D flow, steady and unsteady. Some new image processing techniques and algorithms have been developed for this system. So the visualized 2D flow image can be input automatically, and can be processed under the model directed mode to get the instantaneous distribution of velocity, stream function and pressure with reasonable accuracy. The measurement error of the velocity calculated from the flow visualization picture was within 6%.

Acknowledgements

We wish to thank the reviewers for their useful comments and Mrs. G. Conarroe for the help to prepare this text.

References

1. Adrian, R.J., "Multi-Point Optical Measurements of Simultaneous Vectors in Unsteady Flow – A Review", *Intern. Journal Heat and Fluid Flow*, Vol. 7, No. 2, 127-145 , 1986.
2. He, Z., et. al., "Two Dimensional Fluid-Velocity Measurements by use of Digital Speckle Correlation Techniques", *Experimental Mech.*, Vol.24 , 185-203, 1984.

3. Jiu, J., and F. Schmitt, "Water Current Determination by Picture Processing", *ICASP*, 1981.
4. Imaichi, K., and Ohmi, K., "Numerical Processing of Flow-Visualization Pictures-Measurement of Two Dimensional Vortex Flow", *Journal Fluid Mech.*, Vol **129**, 283-311, 1983.
5. Lee, J. S., "Digital Image Processing by Use of Local Statistics", *IEEE Comput. PRIP*, 55-61, 1978.
6. Rosenfield, A. and Kak, A. C., "Digital Picture Processing", 2nd Edition, *IEEE Academic Press*, 1982.
7. Lvine, M. D., "Vision in Man and Machine", *IEEE McGraw-Hill Book Company*, 1985.
8. Paton, K., "Detection of Thin Line Segments", *Proc. of the 4th IJCPR*, 704-706, 1978.
9. Hedlund, M., et al., "A Consistency Operation for Line and Curve Enhancement", *Proc. of PRIP '82*, 93-96, 1982.
10. Zucker, S. W., "Relaxation Process for Scene Labeling: Convergence, Speed and Stability", *IEEE Trans. on Systems, Man and Cybernetics*, Vol **SMC-8**, No. 1, 283-311, 1978.
11. Prager, J. M. and Arbib, M. A., "Computing the optical flow: The MATCH Algorithm and Prediction", *Comput. Vision, Graph. and Image Processing* , Vol **24**, 271-304, 1983.

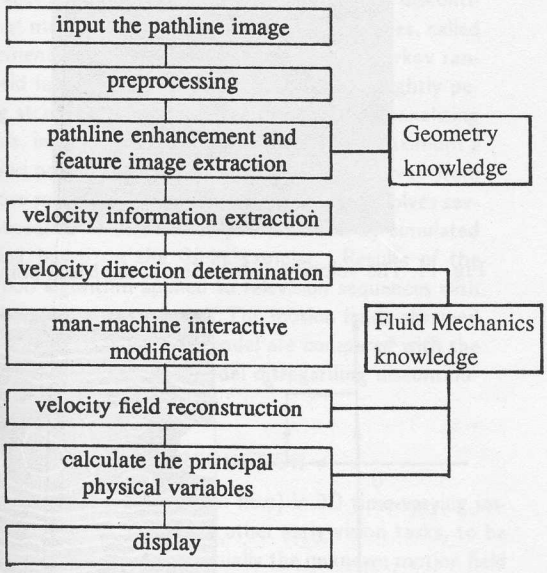
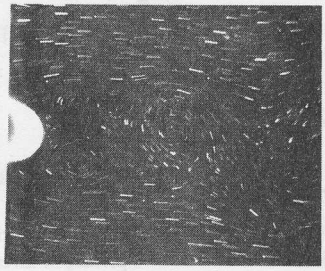
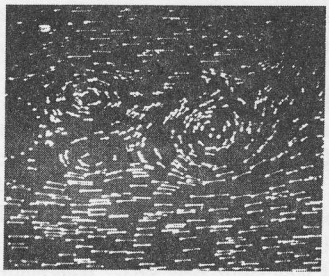


Fig. 1. The major block diagram of our system.



(a) Original flow image



(b) Contrast stretching of (a)

Fig. 2. Preprocessing.

For 0° and 180°

0	A_{11}	A_{12}	A_{13}	0
0	A_{21}	A_{22}	A_{23}	0
0	B_1	B_2	B_3	0
0	C_{11}	C_{12}	C_{13}	0
0	C_{21}	C_{22}	C_{23}	0

For 45° and 225°

0	0	A_{13}	A_{24}	0
0	A_{12}	A_{13}	B_3	C_{24}
A_{11}	A_{22}	B_2	C_{23}	C_{13}
A_{21}	B_1	C_{22}	C_{12}	0
0	C_{21}	C_{11}	0	0

For 90° and 270°

0	0	0	0	0
A_{13}	A_{23}	B_3	C_{23}	C_{13}
A_{12}	A_{22}	B_2	C_{22}	C_{12}
A_{11}	A_{21}	B_1	C_{21}	C_{11}
0	0	0	0	0

For135° and 315°

0	C_{24}	C_{13}	0	0
A_{24}	B_3	C_{23}	C_{12}	0
A_{13}	A_{23}	B_2	C_{22}	C_{11}
0	A_{12}	A_{22}	B_1	C_{21}
0	0	A_{11}	A_{21}	0

Fig. 3. Four templates for initial labeling.

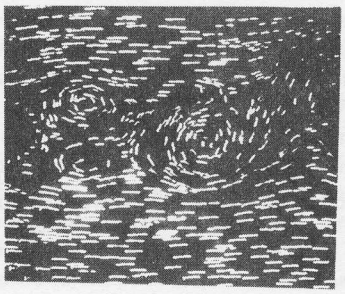
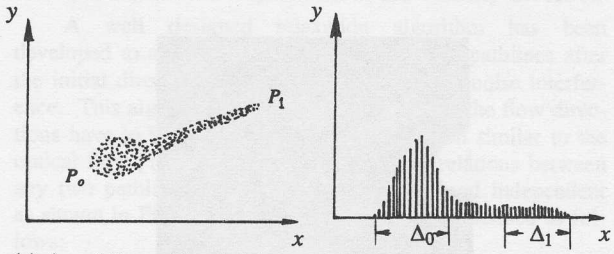


Fig. 4. The enhanced image after 5 iterations.



(a) A pathline with a bright round point. (b) The projection onto the principal direction.

Fig. 5. A sketch of a pathline with a bright round point and its projection.

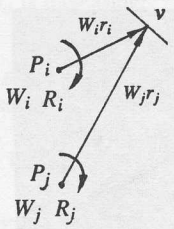


Fig. 6. The comparison between weighted distances. If $W_i = W_j$, it is obvious that $W_i r_i < W_j r_j$.

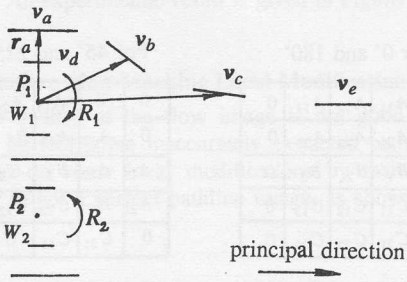


Fig. 7. The direction determination of the velocities in a unsteady twin-vortex flow. The control points are set at the vortex centers. One of the rotation is clockwise and another is counter-clockwise, weights $W_1 = W_2$.

consistent independent unconsistent

Fig. 8. The direction consistency between two pathlines.

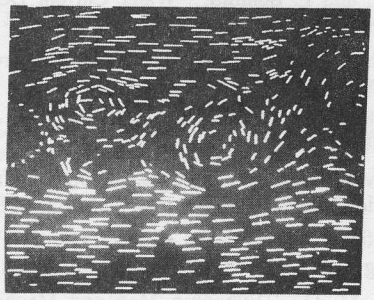


Fig. 9. The pathline extraction.

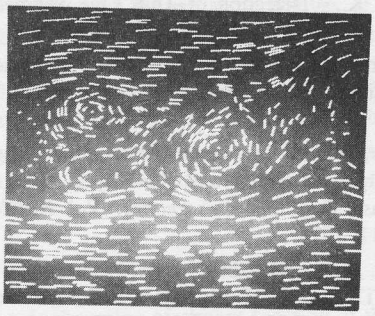


Fig. 10. The flow after the man-machine modification.

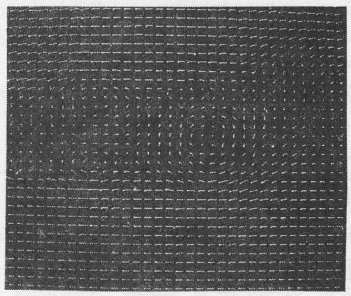
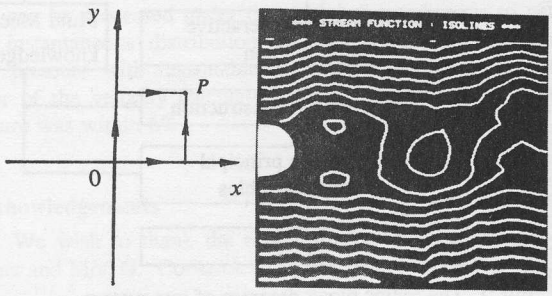


Fig. 11. The velocity distribution at the Mesh points.



(a)The integration path. (b)The stream function.

Fig. 12. Stream function.