This is to certify that the

thesis entitled

Development of a Digital-based Particle Image Velocimetry (PIV) Technique

presented by

Suman Chakrabarti

has been accepted towards fulfillment of the requirements for

M.S. degree in Mechanical Engineering

M. M. Koochesfahani
Major professor

Date 12/20/1991

DEVELOPMENT OF A DIGITAL-BASED PARTICLE IMAGE VELOCIMETRY (PIV) TECHNIQUE

BY

SUMAN CHAKRABARTI

A THESIS

SUBMITTED TO

MICHIGAN STATE UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

DEPARTMENT OF MECHANICAL ENGINEERING

1992

ABSTRACT

"DEVELOPMENT OF A DIGITAL-BASED PARTICLE IMAGE VELOCIMETRY (PIV) TECHNIQUE"

By

Suman Chakrabarti

PIV is a technique for determining the velocity field of fluid flows at many points simultaneously. Images, separated by a time interval Δt, of flows containing seed particles are digitized and processed in pairs. Applying 2-D spatial cross-correlation methods to numerous small areas (or windows) of the images produces the average displacement of all particles within each area. Division of these displacements by Δt yields the respective area-averaged flow velocities.

The above technique was applied to a simulated flow field corresponding to solid body rotation. It was found that a minimum of 4 particles per window was usually sufficient. A maximum out-of-plane particle loss of 20-25% appeared to represent a limiting tolerance. Typically, 90% of all measurements could be determined to 0.06 subpixel accuracy in displacement. The technique was also applied to a wake-shear layer flow. Except for regions of high velocity gradients, the flow characteristics were readily resolvable.

Dedicated to my family,

whose support has been beyond invaluable;

and to the memory of my father,

Mani Lal Chakrabarti

ACKNOWLEDGMENTS

This is to acknowledge the kindness, brilliance, breadth of knowledge, and creativity of my thesis advisor, Professor Manooch Koochesfahani. I grateful for his aid in helping me grow both professionally and personally.

This work has been executed under the Air Force Office of Scientific Research Grant No. AFOSR-89-0417.

TABLE OF CONTENTS

LIST OF FIGURES

NOMENCLATURE

Disclaimer: The term **pixel** is not used in the usual sense herein. Strictly speaking, a pixel is an area. However, there are several parameters which require the use of the linear dimension only, including displacement, pixel coordinates, and correlation peak location. For this reason, the term pixel is assigned the dimensions of **length**. The "unit" of **area** becomes **pixel^2**.

General

Δt	time interval	– between laser pulses or consecutive images
ρ_P	particle density	– no. per interrogation or sample window
s	average particle separation	– vector displacement moved during Δt

Window functions

R	roam window size	
S	sample window size	
W_R	roam window function	– "particle distribution" function
W_S	sample window function	– "particle distribution" function

Grid/pixel coordinates

m	unused image margin	
n	no. samples per grid row	
n_x	x-coord. of sample window	– RE: $n \times n$ sample window grid
n_y	y-coord. of sample window	– RE: $n \times n$ sample window grid
x_c	x-coord. of sample window center point (pixels)	– corresponding to n_x
x_{c0}	x-coord. of 1st sample window center point (pixels)	– RE: 512 × 512 pixel^2 image coordinates
y_c	y-coord. of sample window center point (pixels)	– corresponding to n_y
y_{c0}	y-coord. of 1st sample window center point (pixels)	– RE: 512 × 512 pixel^2 image coordinates

Optical-based PIV (Adrian (1988))

$\tau(X)$	transmissivity of photograph
I	interrogation spot intensity
I_I	interrogation beam intensity
V_I	interrogation volume
X	vector coordinate of interrogation spot
X_I	vector coordinate of interrogation spot center

FFT-based PIV (Willert & Gharib (1991))

$*$	denotes spatial convolution
ϕ	correlation function
$E[\ ]$	expected value
f	sample window function
g	roam window function
s	spatial shift transfer function

2-D Gaussian fitting

C_i	fit coefficients	
x_{MAX}	correlation peak x-location	– to nearest pixel
y_{MAX}	correlation peak y-location	– to nearest pixel
x_{SUB}	subpixel peak x-adjustment	
y_{SUB}	subpixel peak y-adjustment	
x_P	final peak x-location	
y_P	final peak y-location	

Error Processing (*LSD*)

$(E_{FRAC})_q$	velocity comp. fractional error	
F_{LSD}	fraction of velocity grid points that failed *LSD* criterion	
δq	velocity component difference	
$\lvert \delta q \rvert$	absolute value of δq	
N_{SDF}	primary standard deviation factor	
N_{SSD}	secondary standard deviation factor	
q	represents either u or v	
q_{AVG}	local average of velocity comp.	– Calculated from surrounding points
q_{LSD}	local standard deviation (*LSD*) of velocity component	
q_{MEAS}	measured value	– RE: either simulation or real flow
q_{THEO}	theoretical value	– Derived from theoretical equations
$\langle q_{LSD} \rangle$	average of q_{LSD}	– Executed over entire velocity field
$\langle \sigma_{LSD} \rangle_q$	standard deviation of q_{LSD}	– Executed over entire velocity field

Probability functions

δX	horz. displacement difference	– obtained from δu, Δt
δY	vert. displacement difference	– obtained from δv, Δt
$p_{\delta X}(\xi)$	probability x-density function	
$p_{\delta Y}(\xi)$	probability y-density function	
$p(\xi)$	denotes either or both $p_{\delta X}$ & $p_{\delta Y}$	
$P_{\delta X}(\xi)$	probability x-distribution function	
$P_{\delta Y}(\xi)$	probability y-distribution function	
$P(\xi)$	denotes either or both $P_{\delta X}$ & $P_{\delta Y}$	
$\mathcal{P}\{\ \}$	probability	
$X_{90\%}$	ξ corresponding to $P_{\delta X} = 0.9$	
$Y_{90\%}$	ξ corresponding to $P_{\delta Y} = 0.9$	
ξ	displacement error variable	– RE: probability functions
$\xi_{90\%}$	higher of $X_{90\%}$ and $Y_{90\%}$	

Circulation and Vorticity

$\bar{q}$	linear average of u or v between adjacent nodes	
Γ	circulation	
Γ_{MAX}	maximum circulation	
Δl	velocity node spacing	
ω	vorticity	
E_{LOCAL}	circulation error relative to theoretical local circulation	
E_{MAX}	circulation error relative to maximum circulation	
n_r	radius in (n_x,n_y) coordinates	
r	radius (pixels)	
$\bar{u}_x$	path-averaged u-component over x-direction	– RE: horz. portion of $\Gamma(r)$ $\oint$ contour
$\bar{v}_y$	path-averaged v-component over y-direction	– RE: vert. portion of $\Gamma(r)$ $\oint$ contour
ΔL	total path length	– RE: either horz. or vert. portion
$\bar{u}$	path-averaged u-component over θ-direction	– RE: circular segment of $\Gamma(r)$ $\oint$ contour
$\bar{v}$	path-averaged v-component over θ-direction	– RE: circular segment of $\Gamma(r)$ $\oint$ contour

Simulations *(solid body rotation)*

Ω	angular rotation rate
I	particle intensity
I_0	particle peak intensity
σ	particle Gaussian radius
ρ_{64}	particle density / 64 × 64 area
F_ρ	fraction of particles "moved" through 3-D motion

Particle intensity variation

$I(z)$	laser sheet intensity	
$I_{1,2}$	1st/2nd image particle intensities	
w	velocity w-component	
$\hat{w}$	normalized w-component	
z	transverse coordinate	– normal to (and within) laser sheet
$\hat{z}$	normalized transverse coordinate	
z_0	laser sheet thickness	
$z_{1,2}$	1st/2nd image particle locations	– correspond to $I_{1,2}$
ξ_{MAX}	displacement error cutoff value	– used in w-component histograms
Σ_ξ	w-component histogram sum	– sum is over all $\xi \leq \xi_{MAX}$

INTRODUCTION

Particle Image Velocimetry (PIV) is a technique for obtaining the multi-point velocity field of a fluid flow. Given images of a flow containing some sort of passive particles, a PIV system will attempt to track the trajectories of localized groups of particles between the images. These images may be obtained by any of several photographic processes, producing different types of images on varying media. Some examples of specific techniques follow.

Typically, a thin laser sheet is used to illuminate fluid flows. For example, one can utilize a laser sheet that pulses twice (or more) with a time interval of Δt between pulses. The resulting photograph is a double exposure, having captured "images" of the particles at their initial and final locations. The film (like a transparency) is then optically analyzed or "interrogated" by a probing laser to determine the average spacing between the particle pairs within an *interrogation spot* on the photograph (which, in turn, corresponds to an *interrogation volume,* V_I, in the plane of the flow). There are several different methods of determining this average spacing as a result of such a process; Adrian (1986) gives a thorough review of them. Division of the particle spacing by the time interval Δt yields the spatially averaged velocity over that region of the flow. Repetition of this process with a grid of interrogation volumes (or spots) throughout the flow field (or photograph) yields the entire velocity field.

Note the implicit mathematical statement that particles within a sufficiently small

region move at strictly the same speed. Accordingly, the focus of PIV is to determine velocity vectors by following groups of particles in the smallest possible regions (although not mathematically small). The intra-particle distances would then remain *nearly* constant, making it easier to track the particles from initial to final locations. Adrian (1991) makes an explicit distinction by referring to this as *High Image Density PIV*. This implies that there are enough particles within the interrogation volume to make up a unique grouping (ie., one that will only correlate well with itself at future times). Naturally, this assumption becomes invalid in flow regions with high velocity gradients; the information on the changes in velocity would be lost since the PIV technique only resolves a spatially averaged velocity.

It appears desirable to have as many particles as possible within the volume, since it makes the distribution of particles within a region more distinctive, thereby improving the probability of determining the true average particle displacement. An obvious limitation here is that the addition of too many particles would result in a two-phase flow. Another impediment arising from excessive particle density is the possible introduction of false correlations that would obscure the correlation corresponding to the true displacement.

In fact, it is possible to have so many particles that their images (on the double or multiply exposed photograph) overlap one another. Upon illumination by a probing laser, these overlapping particle images produce interference fringes known as *laser speckle*. In this high particle density regime, the technique is referred to as Laser Speckle Velocimetry (LSV) instead of PIV. There are different methods of analysis for LSV: several are reviewed in Adrian (1986), which also discusses other multi-point

velocity field measurement techniques besides LSV and PIV. History and descriptions of PIV are also provided in Adrian (1991), Lourenco, Krothapalli & Smith (1989), and Willert & Gharib (1991).

One of the major underlying causes of error in PIV (and related methods) is loss of particles within the interrogation volume. Possible reasons for the losses are: excessive in-plane particle displacements; excessive out-of-plane particle displacements; or simply insufficient overall flow particle density. High in-plane displacements mean that some of the particles may have sufficient velocity to move past the boundary of the volume, V_I. Large out-of-plane displacements may cause particles to partially or totally disappear from the plane of view (object plane). Clearly, some minimum particle density is necessary both to assure good correlation and to withstand these effects (ie., even if some particles are lost, enough will remain).

Lourenco & Krothapalli (1987) suggest $\rho_P = 4$ (particle density or particles per interrogation volume) as a threshold. They use the Young's fringes optical technique; and so this represents a minimum necessary for good fringe quality (see Adrian (1986) or Lourenco & Krothapalli (1987) for further information regarding Young's fringes). They also state that "this number can be somewhat relaxed if more than two exposures were used for the photography."

Keane & Adrian (1990) state, however, that this criterion is only useful in multiple-pulsed systems with at least five pulses (exposures). This paper continues a theoretical analysis of a double-pulsed PIV system discussed in Adrian (1988), including a series of simulations to test the theory. Figure 1 redisplays the experimental setup from this paper for reference. Six dimensionless parameters are defined therein to

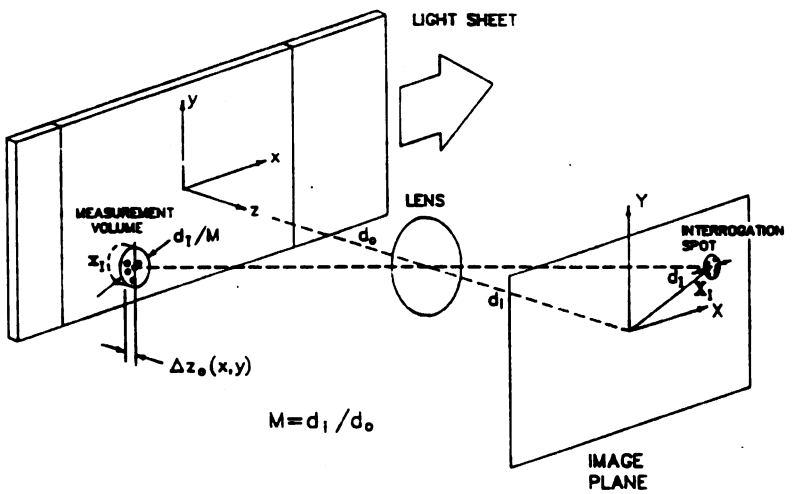


Figure 1. Light sheet and image recording system for planar pulsed laser velocimetry.

Figure 1. Experimental schematic from **Keane & Adrian (1990).**

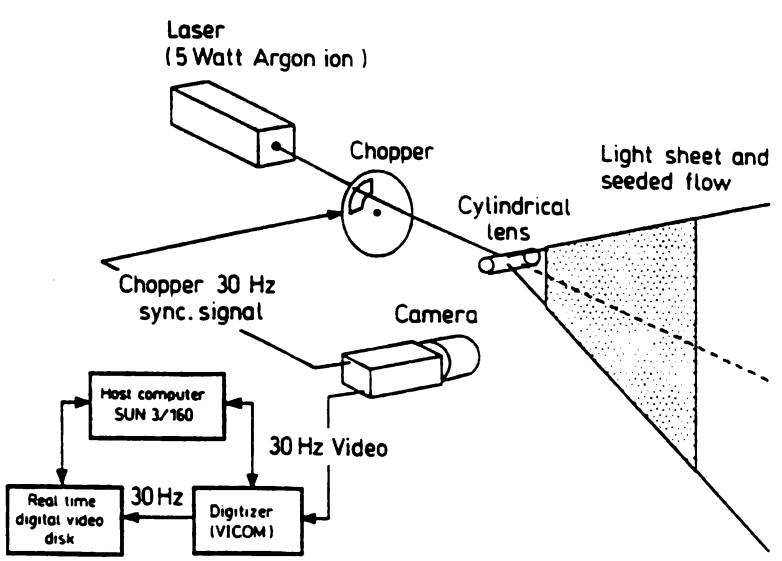


Fig. 8. Schematic of the illumination and imaging system for the DPIV image generation

Figure 2. Experimental schematic from **Willert & Gharib (1991).**

characterize the performance of a PIV system. Recommendations are made for optimal values of these parameters; these include suggesting a minimum of $\rho_P = 10\text{-}20$ particles per volume. Although our implementation is not based on a double-pulsed system, some comparisons can be made based on these parameters; these will be expanded upon in later sections as appropriate.

Willert & Gharib (1991) also performed simulations as well as experiments to establish the capabilities of PIV, albeit in a completely different context. In this case, two images separated in time are generated and digitized (as opposed to one photograph with two particle images for each particle). These images are then analyzed computationally, rather than optically as is done in the previous techniques. For reference, the experimental schematic from this paper is redisplayed in Figure 2. This version of PIV can therefore be considered a digital counterpart of the optical versions (ie., digital PIV or DPIV). A discussion of the limitations encountered by the optical-based techniques together with how they are mitigated by their digital-based technique can be found in the Introduction to Willert & Gharib (1991). These will be briefly summarized in §1.1. This version of PIV is also the basis of work discussed in Fincham, Blackwelder, & Spedding (1991).

Our implementation of PIV is closest to that of Willert & Gharib (1991). A major difference involves the specifics of the digital processing. Willert & Gharib use 2-D FFT techniques to compute the *spatial cross-correlation* of the two images to obtain the velocity data. Our method computes the cross-correlation directly: we use a *sample window* (from the first image) of semi-arbitrary size S, and a *roam window* (from the second image) of size R, where $R > S$. We then perform a direct spatial correlation of

the two windows to obtain the averaged displacement for that sample window (the sample window can be considered analogous to an interrogation spot). This application is similar in vein to that mentioned briefly by Adrian (1991), which uses the terminology *first interrogation window* and *second interrogation window.*

In Willert & Gharib's case, the two windows are the same size: 32×32 pixel^2 ($R = S = 32$), for reasons to do with the nature of an FFT (this will be expanded upon in §1.1). We have used a variety of sizes for both R and S, and our definition of a grid of sample windows also differs. Details of these variations will be discussed in §1.2.

The remainder of the thesis is divided into three sections. Part I describes the principles upon which the software is based. Included are: the use of two-dimensional Gaussian fits to resolve particle displacement; an error detection scheme based on examination of the local standard deviation (*LSD*) of velocity measurements; the filtering routines based upon the previous scheme; and the methods of calculation of circulation and vorticity. Readers interested in applications may wish to proceed to Part II, referring to Part I as needed.

Part II details the results of application of our implementation of PIV. Diverse simulations were executed upon image pairs generated to reflect a solid body rotation flow field. This velocity field was chosen for its inherently large dynamic range and uniform vorticity. The effects of the following were analyzed: variation of particle density, ρ_P; random out-of-plane removal/addition of particles; and variation of particle intensity (which can also represent out-of-plane motion). A real wake-shear layer flow is also examined, with its results considered in the context of the simulation results. When appropriate, all results are also compared with those of Lourenco, Krothapalli &

Smith (1989), Keane & Adrian (1990), Willert & Gharib (1991), and Fincham, Blackwelder & Spedding (1991).

Part III discusses the conclusions and suggests possible improvements that can be made in the software. Also suggested are some guidelines for the application of PIV. All figures pertaining to the material of Parts II & III are included after Part III.

Part I. FEATURES OF TECHNIQUE

1. VELOCITY FIELD PROCESSOR

1.1. Background

Digital vs. optical-based PIV

As was mentioned in the Introduction, Willert & Gharib (1991) make several comparisons between PIV implementations based on fully digital and "opto-mechanical" analysis techniques. Briefly stated, their conclusions are that optical/mechanical analysis can be tedious and time consuming; further, there are inherent limitations in determination of velocity vectors small in magnitude, and that the causality information is lost because these techniques only use a single photographic image. While there apparently exist techniques that can compensate for both of these inherent limitations, they would appear to add complexity to the experimental setup (Willert & Gharib mention several papers covering such enhancements).

Willert & Gharib's digital implementation addresses the above concerns by performing the analysis computationally and using a pair of single-exposed video images. The latter preserves phase information and makes it possible to resolve displacements of particles that moved such a small distance that they partially overlap their original locations in the previous image. Nevertheless, there are at least two limitations to this operational mode of PIV. First, a true photographic transparency offers much better resolution at the present time than 512×512 pixel^2, which is a video standard (Adrian (1991) also discusses this). Second, Willert & Gharib had the 30 Hz (video) framing rate

for their experiments. Such a relatively low image acquisition frequency limits the resolvable velocity range to that of low speed flows (typically of velocity order < 10 cm/s). It should be understood that this aspect of PIV is undergoing rapid evolution: there apparently exist experimental procedures which have the potential for increasing the effective framing rate.

Both Adrian (1991) and Willert & Gharib (1991) note that these are technological constraints, which will ease with further development. Indeed, Adrian mentions the availability of 2048×2048 pixel^2 image arrays, as well as cameras with framing rates of 80 kHz or higher. It is observed that such equipment may not be of universal utility, with high cost being one mitigating factor at present. Another limitation is that very high speed cameras apparently cannot currently produce enough frames to examine a flow field over a long time. Also, the high-resolution arrays are evidently relatively slow. Once again, it should be realized that imaging technology is rapidly advancing at present, presumably resulting in improved speed and performance in the foregoing equipment.

In sum, there is (currently) a tradeoff between speed and resolution when considering digital- vs. optical-based PIV. Adrian (1991) notes that advances in electronics can only expand the possible areas of application of digital PIV. In particular, it is possible that higher framing rates will drastically reduce the various particle losses referred to in the Introduction.* That is, any particle would be given less opportunity to exit the interrogation window or fade from the viewing plane, if permitted less time in which to do so. The ability of digital PIV to resolve velocity vectors from

*That is, excessive in-plane or out-of-plane particle displacements, or insufficient overall particle density.

very small particle displacements — without the additional optical-based experimental enhancements mentioned by Willert & Gharib (1991) — should also be of increased utility with faster cameras.

Our differences from Willert & Gharib (1991)

We have adapted our technique from that of Willert & Gharib (1991), with several modifications. As remarked upon in the Introduction, our sample and roam window sizes (R and S, in pixels) differ because of the method by which we compute the cross-correlation. We do it directly (also, Adrian (1991) refers to it as *direct cross-correlation*), while Willert & Gharib use two-dimensional FFT's. Since the FFT is applied to data sets consisting of 2^n points (Press, Flannery, Teukolsky & Vetterling (1986)), one must either set R and S to a number of pixels equal to a power of 2, or "pad" the windows with zeroes to fill them out to such a number. The latter naturally would add noise to the process, perhaps requiring some sort of filtering. Willert & Gharib avoid this entirely by simply setting $R = S = 32$.

Our implementation does not necessitate such a restriction upon window size. We have been able to vary both R and S in a semi-arbitrary fashion. Our main limitation is self-imposed: since it is easier to programmatically define a window's center if its size is an odd number (in pixels), we have explicitly made R and S odd. Currently, our software permits size ranges as follows:

$$\begin{aligned} 7 < S < 51 \ &pixels \\ R \leq 85 \ &pixels \end{aligned} \qquad (1)$$

The above dimensions are in the context of 512 × 512 pixel^2 images; both we and

Willert & Gharib (1991) use video cameras to digitize images at the standard 30 Hz framing rate.

Willert & Gharib also emphasize the use of two 1-D gaussian fits to determine the average particle displacement. These are performed around the highest correlation peak resulting from the spatial correlation of the two windows. They state that this method of obtaining the peak yields subpixel accuracy and is more accurate than the more "traditional" method of centroiding (Adrian (1988), Keane & Adrian (1990), and Adrian (1991) all discuss centroiding in varying degrees of detail). The basis of this conclusion apparently arose from a series of experiments using both techniques. We have carried this statement farther by using a single *2-D gaussian fit* to obtain the subpixel location of the correlation peak. We expect that this should further reduce the uncertainty of the fitting process.

We further do not *deliberately* use seed particles in our interrogation of fluid flows, whereas Willert and Gharib (1991) used 80 μm phosphorescent spheres. Lourenco & Krothapalli (1987) also use spherical "tracer" particles in their application of PIV; they establish several guidelines for particle size as well as other experimental parameters. Adrian (1991) states that actual "particles" for use in PIV may in fact be solid, liquid, or gaseous, depending upon the medium in which they move. In our case, the particles used were simply the naturally occurring dirt grains in the local (Michigan State University) water.* It is believed that they are passive, with negligible buoyancy. The drawback of this PIV mode is that some degree of image pre-processing becomes

*Flows with hydrogen bubbles as particles are also under current investigation.

necessary to enhance these nonuniform particles. Willert & Gharib (1991) explains that no image pre-processing such as thresholding was needed for their images because of the sufficiency of contrast between their flow medium and phosphorescent particles. It is probable that such techniques would enhance the contrast between our nonuniform particles and the flow background. At this juncture, our development of image pre-processing procedures is in the preliminary testing stage; some concepts under current investigation will be discussed in Part III. It was nevertheless possible to analyze real flow images even *without* such enhancement. We also expect that our PIV system would even more readily process images with high "quality" particles such as those of Willert & Gharib (1991).

We performed extensive simulations on artificially generated pairs of images to establish the effects of some important parameters (simulating particles of reasonable "quality"). These image pairs were designed and constructed to reflect a solid body rotation motion on the part of the particles. As stated in the Introduction, this flow field was chosen for its wide variation in velocity (in our case, from 0 at the lower left corner to a maximum at the largest radius) and uniform vorticity. We have obtained several different estimates of the upper limit of performance of our PIV implementation, as well as the effects thereon of several sources of loss/noise. The analysis of these simulations is presented within §3.

Details of spatial correlation will be presented in mathematical and graphical form. Comparisons will be made with the extensive theoretical calculations in Keane & Adrian (1990), as well as the fully digital formulation of Willert & Gharib (1991).

1.2. Derivation of data window functions

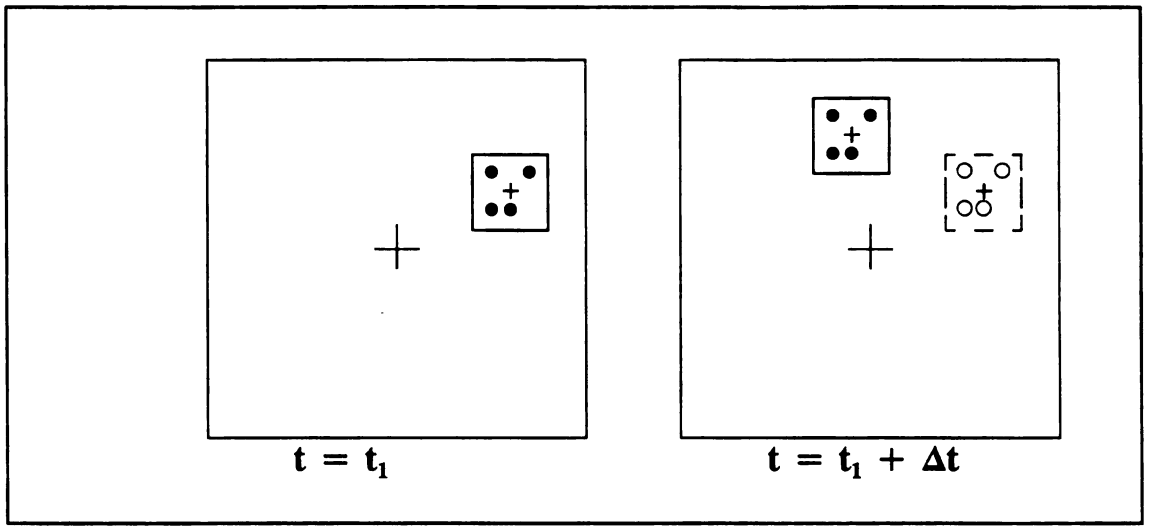


Figure 3. Image pair with consecutive locations of a group of particles.

Figure 3 shows two images separated by a time interval Δt with portions (windows) containing a finite number of particles. The intra-particle spacing remains roughly constant as the entire group moves an average distance Δs between the first and second images. This distance can be observed from the separation between the center points of the windows. In fact, this center point separation is exactly the quantity measured by the software to determine Δs, first to the nearest pixel, then to subpixel accuracy.

First, a grid of sample windows is defined as shown in Figure 4. They are directly adjacent to one another, as well as non-overlapping (overlapping of sample windows will be retained as an *optional* capability). That is, the distance between center points of two consecutive sample windows of size S is also equal to S. This differs from Willert & Gharib (1991) in that they step their 32 $\times$ 32 window in increments of 8 pixels. This relative redundancy of information apparently permits them to take a

straight low pass filter of their data as a part of their post-processing. In our case, the two window sizes, R and S, determine the total number of windows on the grid. The initial values for the number of sample windows in one row of the grid, n, is

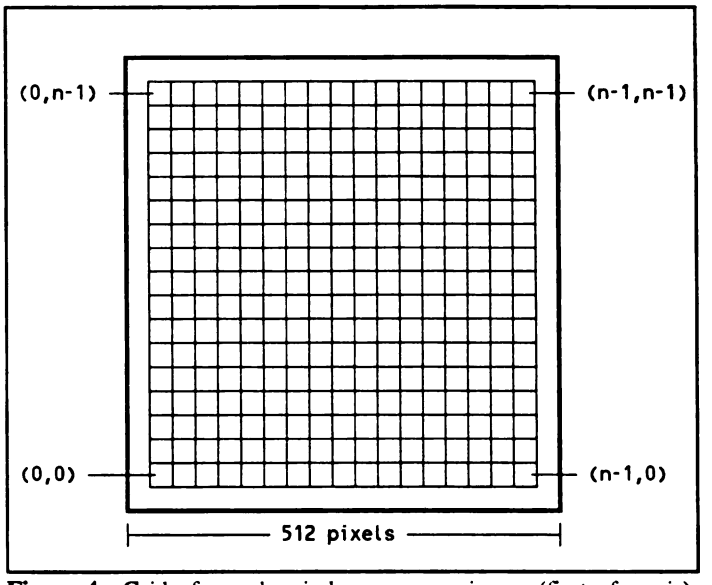


Figure 4. Grid of sample windows over an image (first of a pair).

$$n = \frac{512}{S} \; ; \; n \in \{ \textit{integers} \} \quad (2)$$

Figure 4 therefore depicts an $n \times n$ sample window grid. Each roam window is emplaced concentric with its corresponding sample window, and is therefore larger by an equal number of pixels in each direction (see Figure 5). Willert & Gharib (1991) mention that the Nyquist sampling criterion (with respect to FFT's) restricts their maximum recoverable displacement to $S/2$ pixels for an interrogation window of size S. They further state that attempting to resolve even this displacement often resulted in the

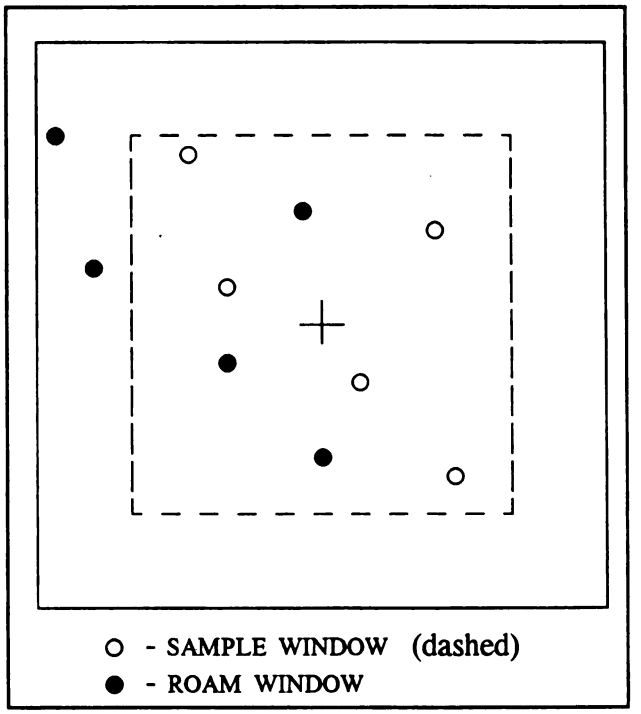


Figure 5. Superposition of sample and roam windows containing idealized particles.

addition of noise to the cross-correlation. This last arises from the increased probability of random particles of the roam window matching those within the sample window. The limit chosen on basis of their experience was *S/3*. Note that for their value of $S = 32$, this represents a ceiling of about 10-11 pixels. We decided to use this limit as an initial guideline and to provide a basis for comparison. Thus, in our case,

$$R = S + 2\frac{S}{3} \tag{3}$$

Note that the maximum values from equation (1) are selected to correspond to equation (3). The value of R given by equation (3) eventually served as a minimum, subject to increase for a variety of reasons (discussed in §4.1.1 and the Conclusions). Note that we are not restricted by the Nyquist criterion in the same sense as in Willert & Gharib (1991) (ie., in a non-FFT environment). Nevertheless, it certainly remains true that one should not *blithely* assign larger and larger roam windows for the same reasons regarding the addition of noise to the process of correlation.

The division in equation (2) always leaves a remainder (because S is always odd); this fractional part is converted into the unused margin of the image as follows:

$$m = \frac{MOD(512,S)}{2} \tag{4}$$

where $MOD(p,q)$ is the modulus function, returning the remainder of the division of p by q. In fact, this formulation of n and m needed to be modified for slightly larger values of m. Otherwise, interrogation of sample windows located at the edge of the image (ie., considering Figure 4, those windows with an "x" or "y" value of 0 or $n-1$) was hampered by the possibility that the corresponding roam windows might not be

entirely on the image (cf. Appendix §A.1 for details).

1.3. Spatial cross-correlation

The following consists of a discussion of PIV processing in three different contexts: optical cross-correlation in a double-pulsed laser system; FFT-based digital cross-correlation; and direct digital cross-correlation using the sample and roam windows from §1.2. The digression at this point from the material of §1.2 to these two prior implementations of PIV is done to place our methodology in an established framework and serve as a basis for comparison (readers may prefer to proceed directly to §1.3.3 for information pertaining to our own implementation only).

1.3.1. Optical cross-correlation

The process of spatial cross-correlation is discussed in Adrian (1991) in several contexts. Most of the discussion is on optical-based PIV, with a reference to a possible digital-based system. This being the case, Adrian delineates a correlation formula — for a double-pulsed laser system—based upon the intensity of the laser beam interrogating a spot on the photographic transparency $I_I(X-X_I)$ and the transmissivity of the photograph, $\tau(X)$. The vector coordinate X is defined within the interrogation spot (ie., the "image" plane), with X_I denoting the center of the spot. Then the interrogation spot intensity is

$$I(\vec{X}) = I_I(\vec{X}-\vec{X}_I)\tau(\vec{X}) \qquad (5)$$

This can be interpreted as the intensity directly "after" or "behind" the photograph (Adrian (1988), Keane & Adrian (1990), Adrian (1991)). This intensity is then

correlated with itself over all possible displacements, s, over the interrogation spot as follows

$$R(\vec{s}) = \int_{spot} I(\vec{X})\, I(\vec{X}+\vec{s})\, d\vec{X} \qquad (6)$$

The correlation function R thus represents the correlation of the beam intensity with a displacement s, as a function of s. Technically, R is called an *autocorrelation* function, since the intensity function is being correlated with itself, albeit at a later time (Rosenfeld & Kak (1982)). It appears therefore that R should have at least two locations with high correlations: at $s = 0$, which indicates zero displacement or the particles within the interrogation spot are correlating ("matching") with themselves; and at the value of s corresponding to the actual distance moved by the particles.

Suppose, for example, that ten particles are within a particular interrogation volume of a flow; then — for a double-pulsed system — twenty particle images would be contained within the interrogation spot (provided that none were lost). At $s = 0$, there would be a correlation peak corresponding to roughly twenty images matching with themselves. Adrian calls this the self-correlation or pedestal component of R, R_P. For the s value representing the true distance moved, there would be a peak about half the size of R_P, since there are only ten particle images matching. That is, the ten particle images at the initial locations within the spot match with the ten particle images at the final locations. Note that the reverse can also be true: the final locations can match with the original locations at an s value that is the exact opposite to the previous value. Adrian therefore terms these two values of s as the displacement correlation peaks R_{D+}

and R_{D-}.

Thus, the magnitude of the displacement is established by either displacement peak, but the direction is not. This is the directional ambiguity referred to by Willert & Gharib (1991), which mentions that additional enhancements to the experiment are necessary to resolve that. They mention an example of such an enhancement, called *image shifting*, from a paper by Landreth & Adrian (1988). Lourenco et al (1989) also discusses image shifting extensively. The basic principle of the technique is to experimentally perform an optical Galilean transformation on the velocity field. This addition of a constant velocity resolves the ambiguity by biasing the measurement of the displacement peak. The source of the ambiguity, however, is the basing of the measurement upon a single image, as mentioned in the Introduction.

1.3.2. FFT-based digital cross-correlation

Willert & Gharib (1991) discuss the cross-correlation process in the context of data window functions instead of intensities (Adrian (1988) also mentions this parallel). They first state that the process can instead be expressed as a *convolution* involving the two window functions and a spatial shift *transfer function*, together with noise,

$$g(i,j) = f(i,j) * s(i,j) + (\text{noise}) \qquad (7)$$

where f and g are the sample and roam window functions, s is the spatial displacement (Willert & Gharib also term it the "impulse response"), and * represents the spatial convolution of f and s. More explicitly, the convolution of the initial particle distribution

function f with a spatial shift s produces a new particle distribution function g.* Willert & Gharib further state that the convolution process can be expedited if one neglects the noise effects and takes the Fourier transform of equation (7); division and deconvolution then yield s.

This method is apparently very sensitive to noise indeed. Therefore, they instead enact a spatial cross-correlation of f and g as a statistical expected value E:

$$\phi_{fg}(i,j) = E[f(i,j),g(i,j)] \tag{8}$$

The above equation can be rewritten in discrete form in the following fashion (see Willert & Gharib (1991) for details of derivation):

$$\phi_{fg}(i,j) = \frac{\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} f(k,l)\,g(k+i,l+j)}{\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} f(k,l)\;\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} g(k,l)} \tag{9}$$

where the denominator is simply a normalization of the correlation function. If noise can be neglected, the substitution of equation (7) into the numerator of equation (9) produces the following sequence:

$$\begin{aligned}\phi_{fg}(i,j) &= E[f(i,j),g(i,j)] \\ &= E[f(i,j),f(i,j)*s(i,j)] \\ &= \phi_{ff}(i,j)*s(i,j)\end{aligned} \tag{10}$$

where Willert & Gharib recognize ϕ_{ff} as the autocorrelation function of f. Note especially the similarity in form of equations (6) and (10). This emphasizes that these are two methodologies of the same basic technique.

---

*Note that there exists a difference in meaning between "convolution" and "correlation." According to Press et al (1986), a convolution processes a signal with its response, whereas a correlation processes two (usually similar) signals.

1.3.3. Direct digital cross-correlation

Willert & Gharib (1991) explain that they speeded up the cross-correlation process through the use of FFT's. Indeed, they assert this fact as the most important feature of their implementation. Our implementation is also digital-based, but there was no need for such a measure: we simply calculate the correlation function ϕ directly. Consider the sample and roam windows of §1.2 in terms of two-dimensional data window functions:

$$\begin{aligned} \textit{SAMPLE}&: \quad W_S(i,j) \;\; ; \;\; i,j \in [1,S] \;\; (\textit{pixels}) \\ \textit{ROAM}&: \quad W_R(i,j) \;\; ; \;\; i,j \in [1,R] \;\; (\textit{pixels}) \end{aligned} \tag{11}$$

These functions may also be thought of as random particle distributions. The intervals [1,S] and [1,R] represent the *local* pixel coordinates with respect to lower left corner of each window, which is considered to be [1,1]. The windows' common center point is referenced to the overall 512 × 512 pixel^2 image coordinate system.*

With the coordinates of all samples defined, we simply performed the cross-correlation of each sample window with its corresponding roam window directly in the space domain,

$$\phi(h,k) = \sum_{j=1}^{S} \sum_{i=1}^{S} W_S(i,j)\, W_R(i+h, j+k)$$

$$\textit{where} \left\{ \begin{array}{c} h,k \in [0, R-S] \\ (\;0\;,\;0\;) = \textit{upper left corner of domain} \\ (R-S, R-S) = \textit{lower right cornerof domain} \end{array} \right\} \tag{12}$$

Note that the formula has already been discretized to resemble the enacted

---

*See Appendix §A.2 for the conversion from grid coordinates to pixel coordinates. Note according to Appendix §A.2 that a roam window's coordinates are defined with respect to the same (center) point as a sample window's.

programmatical procedure. Further, the functions have been defined such that there is a constant offset of $(R-S)/2$ in displacement measurement in both the horizontal (u-component) and vertical (v-component).* A final note regarding equation (12) is that $\phi(h,k)$ is as defined for a specific grid point or (n_x,n_y) pair. Thus, one can loosely consider ϕ to be of four-dimensional nature.

1.4. Velocity determination

Correlation peak analysis

As stated in §1.3.1, the result of a cross-correlation based on a double-pulsed PIV system is a functional domain with a self-correlation peak and two displacement correlation peaks. The nature of this correlation domain is different in a single-pulsed system such as that of Willert & Gharib (1991) or ourselves. Because there are two distinct images, there is no self-correlation peak. Further, there is only a single displacement peak. In either case, the goal of PIV is to make a precise measurement of the location of the peak (presumably using some appropriate technique to determine which displacement peak is the true one in a double-pulsed system). Recall that there remain other methods to determine displacement, including the Young's fringes method discussed in Adrian (1986), Lourenco et al (1989), Keane & Adrian (1990), and Adrian (1991).

---

*It became apparent that it was much faster to simply subtract this constant from the measured u- and v-components after the 2-D gaussian fit, rather than account for it during the correlation process. Another programming alternative would have been to set the indices of the array variables containing the window functions and correlation domain to be symmetric about zero; this also would have entailed additional bookkeeping.

Keane & Adrian (1990) determine the peak's location through finding its centroid. Willert & Gharib (1991) use three-point exponential curve fits to do the same; their claim of superior accuracy over the center-of-mass centroiding technique is evidently based on supplemental experiments comparing these two techniques along with a parabolic fit. They attribute this superiority to the roughly Gaussian shape of the correlation peak.

Subpixel peak location

Our implementation, like Willert & Gharib's, finds the correlation peak initially to the nearest pixel. As previously described, we then employ a two-dimensional Gaussian fit over a 3 × 3 pixel^2 correlation "window" centered upon the peak (pixel) location. The coefficients to be determined are contained in the functional form,

$$F(x,y) = e^{(C_0 + C_1 x + C_2 x^2 + C_3 y + C_4 y^2)} \qquad (13)$$

The information in the 3 × 3 submatrix is then converted into a 5 × 5 system of equations to be solved for the C_i's. For a thorough discussion of the matrix theory underlying the foregoing, see the pages indicated with the reference listing for Press et al (1986).

The subpixel location of the peak is then found by determining the relative maximum of $F(x,y)$ (see Appendix B for explicit formulae). Note that the actual displacement is realized by subtracting the constant offset mentioned regarding equation (12) from this relative maximum. The resulting displacement is divided by Δt to yield the subpixel velocity measurement. Repeating the correlation/fitting process for the entire sample window grid produces the "raw" velocity field (together with other

supplemental data usable for diagnostic purposes). This raw velocity data becomes the input to most of the rest of the processing software, including: vector plotting; error detection and analysis; filtering; and circulation/vorticity calculation.* These velocity post-processing procedures will be discussed in the remainder of Part I.

*Examples of velocity vector plots are included after Part III, and will be discussed together with other results in Part II.

2. VELOCITY FIELD POST-PROCESSING

2.1. Background

PIV error sources

As has been commented upon, a major source of error in PIV is a shortfall of particle density, ρ_P. The obvious preventive measure against this error is to seed the flow with a sufficiently high number of particles. This does not prevent random fluctuations in ρ_P from sample to sample. These variations may then result in too few particles in some of the sample (or interrogation) windows. The Introduction mentioned criteria recommended by other researchers to minimize these particle density fluctuations. These criteria included restrictions on particle density and both in-plane and out-of-plane particle displacements.

Adrian (1988) and Keane & Adrian (1990) conduct an extensive theoretical study and categorization of these error causes and their effects (much of this is summarized in Adrian (1991)). Keane & Adrian, in particular, propose several definitions and suggest error reduction criteria and terminology. Although these were developed in the context of a double-pulsed laser, optical interrogation-based PIV system, the concepts are stated to be valid and conditionally usable both for PIV implementations and other multi-point velocity measurement techniques such as Laser Speckle Velocimetry (LSV). For example, their nomenclature for excessive in-plane displacement that takes a particle beyond the bounds of a sample (or interrogation) window is the *loss of pairs effect.* This

terminology arises from the fact that a double-pulsed PIV system generates two images for each particle (as referred to in the Introduction); thus one particle image remains within the sample window while the other (later in time) does not, and becomes the missing half of a pair.

Other losses can occur as a result of a *biasing* of the velocity measurement. One example of such a bias is called *gradient bias* by Keane & Adrian (1990). This occurs if the chosen sample window is not small enough to minimize a gradient of velocity within it. The result is a *broadening* of the correlation peak, with particles within the sample window producing correlation peaks that no longer overlap each other to produce a single narrow peak. Since PIV is a spatially averaging measurement technique, the faster part of the gradient is averaged with the remainder. Depending on the strength of the gradient, this may result in a shift of the displacement correlation peak to a lower value than the actual average particle displacement.

A related type of bias is called *detection bias* by Keane & Adrian (1990). This also takes place in flow regions with velocity gradients sufficient to cause variation in particle displacements within a sample window. Recall that the magnitude of a correlation peak is dependent upon the number of particles that match at that displacement value (cf. §1.3.1). The end result is a *lowering* of the correlation peak together with the broadening of gradient bias. A sample window sized to minimize internal gradients (ie., better spatial resolution) will make these biasing effects small. Since the two are interrelated, they are referred to as *velocity bias* in Adrian (1991). Willert & Gharib (1991) also treat them jointly.

Willert & Gharib (1991) further mention a strictly digital consideration: their use

of three points in each direction to simulate the correlation peak for the purpose of a Gaussian fit introduces a certain amount of error. This remains true with our own implementation, although we have reduced the error somewhat by utilizing nine points and using a 2-D Gaussian fit to obtain the horizontal and vertical displacements simultaneously (cf. §1.4). A more general limitation is that PIV cannot easily take curvature of the particle trajectory into account. Such curvature can be minimized with a smaller Δt or increased spatial resolution. It is also possible that an improved algorithm can actually "search" for the magnitude of curvature in a trajectory being measured.

Note that an increased overall particle density permits the use of a smaller sample window, which increases the spatial resolution and decreases the apparent spatial gradients over the size of the window. A higher ρ_p can therefore compensate to some degree for these other error sources; an insufficient density only exacerbates them. An increase in ρ_p is not always possible in a given experiment.* As expressed in §1.1, we do not seed the flow with extraneous particles as a part of our experimentation, instead using the naturally occurring bubbles and dirt particles. It is therefore necessary to develop techniques for testing the veracity of all measurements and correct those that are most questionable due to any of the preceding errors.

General error analysis

Adrian (1991) divides the class of PIV error analysis into two categories: those

*Even if it is possible, an augmentation of ρ_p may not be advisable. One reason already stated would be that an excessive particle density would result in effectively a two-phase flow.

enacted during the processing of the velocity field; and those performed after that processing has been completed. In the former case, the crux is to determine whether the highest correlation peak is the true displacement peak or a noise peak. Such a noise peak can occur if the actual displacement peak is not sufficiently higher than the background level of the correlation domain (which in turn can be attributed to insufficient particle density to produce a higher displacement peak). Adrian states two methods by which to reduce the possibility of a noise peak being the highest: restricting the search for the displacement peak to that region of the correlation function where the velocity vector is *expected* to be; and/or mandating that the size of the highest peak be higher than a *detection threshold.* Keane & Adrian (1990) extend the latter by defining the *detectability, D,* to be the ratio of the tallest peak to the second tallest peak. It is suggested that D should be greater than some D_0 for a valid measurement, where $1.2 < D_0 < 1.5$ is the recommended range for the threshold. Once again, the essence of both of the foregoing procedures is that the measured vectors are scrutinized immediately for validity.

The other class of error processing is to allow invalid vectors during the velocity field processing and replace them afterwards if they are drastically different from their neighboring vectors. Adrian (1991) refers to work by Landreth & Adrian (1989) which carries out this type of procedure. Also mentioned is that several of the next highest correlation peaks can be stored to determine if they are more suitable displacement peaks; this can be done concurrently with the checking of neighboring vectors.

Willert & Gharib (1991) also perform a procedure based upon comparison of a velocity vector with adjacent vectors. The first part of their processing consists of

stepping their sample windows by *S/4* to produce measurements redundant over a 3 × 3 set of vectors. Then, the post-processing involves interactive removal of those points judged to be extremely different from the eight adjacent points immediately adjacent. It was affirmed that only a small percentage of the overall velocity field needed to be replaced in this fashion. Each bad vector is replaced by the average of the 3 × 3 set centered on that vector. Finally, the entire velocity vector field is replaced in the same fashion. It is stated that this has the effect of a spatial low-pass filter "which removes the high frequency jitter associated with the different location estimates of the peak correlation."

Our own procedure is a mostly post-interrogation procedure that is also founded upon the comparison of vectors with the immediately surrounding vectors. This comparison is carried out through basic statistical analysis of the eight vectors adjacent to each central one being investigated. The result is the mean and standard deviation of each eight vector set (henceforth these eight-point properties will be referred to as *local* properties). If the u-/v-components of the measured central vector are sufficiently different from the local mean u-/v-components, then that central vector is targeted for replacement by this local mean. The local standard deviation — or *LSD* — becomes the crucial parameter, since it yields a measure of the size of the aforementioned differences in the u- and v-components (compared to the local mean u and v).

The focus of the remaining sections is to illustrate the statistics, capabilities, and limitations of the *LSD* method. Eventually, the methods of PIV accuracy determination will be discussed. These include techniques to establish the accuracy of the PIV implementation upon a simulated flow field, followed by methods of estimation of the

average fluctuation of measurements of real flows.

Velocity derived quantities

The final section of Part I covers the calculation of velocity-derived quantities. The two quantities evaluated in our study are circulation and vorticity; the capability to determine additional properties is readily available as a future option. Since it is apparently difficult to measure vorticity and circulation directly, an ability to calculate them indirectly provides a useful supplement to velocity field determination. For developmental purposes, our study emphasized obtainment and accuracy of velocity data. Further evolution of the PIV technique — especially in image enhancement — will allow for a greater focus on velocity-derived quantities.

2.2. Error correction: the *LSD* method

2.2.1. Error detection

One note regarding the forthcoming error processing is that some of it applies only to the simulated flows (cf. §3.1). This is done to establish the accuracy of the PIV implementation; therefore all obtained velocities are compared to those given by the appropriate theoretical equations (eg., for §3.1, these would be the equations governing solid body rotation). The difference between measured and theoretical velocity components can be expressed as

$$|\delta q| = |q_{THEO} - q_{MEAS}| \qquad (14)$$

where q can represent either u or v.

Equation (14) can be considered an indicator of *relative* velocity (component)

error. A *fractional* velocity error can also be defined. Taking into account the possibility of zero values in u or v,

$$(E_{FRAC})_q = \frac{|\delta q|}{q} = \begin{cases} 0 \ , & |\delta q| = 0 \ , \\ \frac{|\delta q|}{q_{THEO}} \ , & q_{THEO} \neq 0 \ , \\ \frac{|\delta q|}{q_{MEAS}} \ , & q_{THEO} = 0 \ . \end{cases} \qquad (15)$$

where: $q \in \{u, v\}$

The fractional error $(E_{FRAC})_q$ can provide an insight into the magnitude of velocity error at a specific location, independent of the magnitude of velocity itself. In regions of low velocity, this error may be inflated. Therefore it is not incorporated into the *LSD* method, but nevertheless retained as an alternative diagnostic.

In real flows, it is more difficult to find a standard for comparison for the measured velocity field based on a known analytical solution. Therefore, the average of the surrounding eight u-/v-components (cf. §2.1) were employed as the new basis. The "error" was calculated from the difference between measured and local mean u-/v-components (with q still representing either):

$$|\delta q| = |q_{AVG} - q_{MEAS}| \qquad (16)$$

Equation (16) is therefore an indication of consistency of measured velocity vectors with respect to their neighbors. The essence of the *LSD* method is to provide a criterion for a tolerable level of inconsistency. This criterion is the local standard deviation or *LSD* of the eight grid points (vectors) surrounding the velocity vector being analyzed.

In a more general sense, standard deviation is a measure of the uniformity (or lack thereof) of a list of values. The smaller it is, the less variation exists among

individual values of the list. This also indicates that the average of the list is more reliable in some sense (eg., if all the values of a list are equal, the standard deviation is zero). Calculation of each velocity component *LSD* is based upon the unbiased estimator,

$$\sigma = \sqrt{\frac{\sum (p_i - avg)^2}{n - 1}} \qquad \begin{array}{rl} \textit{where:} & n = \textit{number of items in list} \\ & p_i = \textit{the ith item in list} \\ & avg = \textit{average of values in list} \end{array} \tag{17}$$

This is sometimes referred to as the *sample* standard deviation in a statistics context.

For any grid point or vector, (n_x,n_y), under investigation, each component's *LSD* is defined relative to the following *local* average of those components,

$$q_{AVG} = \frac{\left\{ \sum_{j=n_y-1}^{n_y+1} \sum_{i=n_x-1}^{n_x+1} q(i,j) \right\} - q(n_x, n_y)}{8} \tag{18}$$

$$n_x, n_y \in [0, n-1] \quad , \quad \textit{where } n \equiv \textit{no. grid points per row}$$

Note that the velocity component of the center point (n_x,n_y) is included within the double summation, then explicitly subtracted. Substituting this u- or v-component average into equation (17) then yields that component's *LSD*:

$$q_{LSD} = \sqrt{\frac{\left\{ \sum_{j=n_y-1}^{n_y+1} \sum_{i=n_x-1}^{n_x+1} (q(i,j) - q_{AVG})^2 \right\} - (q(n_x, n_y) - q_{AVG})^2}{8}} \tag{19}$$

Again, observe that equation (19) is similar in form to equation (17) except for the exclusion of the term corresponding to the center point (n_x,n_y).

2.2.2. A "smarter" low pass filter

At this point, the magnitude of the velocity difference, $|\delta q|$ (equation (16)) is compared to the *LSD* for each velocity component (equation (19)) according to the following condition:

$$|\delta q| > N_{SDF} \times q_{LSD} \qquad \textbf{(20)}$$

$$\textit{where: } N_{SDF} \equiv \textit{standard deviation factor}$$

That is, a grid point having a u- or v-component that deviates by more than N_{SDF} standard deviations from the local average, q_{AVG}, is targeted for replacement. N_{SDF} is a constant that can be chosen arbitrarily for a particular flow. This enables adjustment of the sensitivity of the process of filtering and replacement of velocity vectors. We have generally used $N_{SDF} = 2$ for our PIV implementation. Note that this is a relatively conservative criterion, usually isolating roughly the very worst 5-10% of the measured velocity vectors (this grid fraction is denoted F_{LSD}; this definition will be used in Part II). Such a coarse "filter" is desirable here as a *substitute* for interactive identification and removal of the very worst vectors. $N_{SDF} = 2$ appeared to provide a sufficiently coarse level of selectivity, without being so narrow as to fail to identify all of the conspicuously bad vectors.

Willert & Gharib (1991) mentions an interactive step (of bad vector identification and removal) before application of a straight low pass filter to remove the "high frequency jitter associated with the velocity data". Performing the low pass filter without the foregoing intermediate step would result in data contamination: the vectors adjacent to the extremely bad ones would become more erroneous due to the "influence" of the

bad vectors in the filtering.*

The low pass filter remains useful in removing this "jitter" from our own velocity data. Note that in our case, the mechanism of bad vector detection can become **nearly automatic** given an appropriate choice of N_{SDF}.† Conversely, the technique may fail in cases of several adjoining bad vectors. Recall from equation (19) that the velocity component *LSD's* are calculated based on the surrounding eight vectors. If any of these vectors are in error, then the *LSD* of the grid point under inspection is artificially inflated (as mentioned before, standard deviations are a measure of the *uniformity* of lists of values). Isolated regions of such bad vectors would thereby produce higher *LSD's* within those regions. Unless the vector being examined has a very large $|\delta u|$ or $|\delta v|$, it will not be identified as a bad vector by the equation (20) condition (this condition can also be termed the "*LSD* criterion"; the identification of a vector by equation (20) may therefore be labeled as the vector's "failing the *LSD* criterion" or similar phrasing).

It must be emphasized that failure of identification of bad points by the *LSD* method occurs mostly with closely adjacent clusters of bad grid points. Further, it has been observed that if only two or three bad points are adjacent, then they do not *always*

---

*Note that this low pass filtering process consists of replacement of every velocity grid point with the average of the (usually eight) surrounding grid points.

†Even the targeting of mildly erroneous vectors is not unduly harmful for three reasons: 1) $N_{SDF} = 2$ is apparently sufficiently coarse to ensure that such targeting is uncommon; 2) the replacement of such a vector by its local average (q_{AVG}, based on the surrounding vectors as shown by equation (18)) will not introduce significant error unless at least some of the neighboring vectors *are* in fact grossly erroneous; 3) the previous is rendered even less likely by the procedure of *sorting* the list of bad vectors in order of decreasing velocity differences ($|\delta q|$) *before* replacing these vectors (this strategy is described on page 34).

It is therefore more probable that one of these hypothetical neighboring grossly erroneous vectors will be replaced first, meaning that the local average of the moderately erroneous vector will not be contaminated by that extremely bad vector. This erroneous vector simply will no longer exist, because of its earlier replacement by its own local average.

fail to be identified by the *LSD* criterion. Nevertheless, interactive selection similar to that of Willert & Gharib (1991) appears to be needful in such circumstances to complete the task begun by the *LSD* method. At this point in the development of our PIV system, such an interactive step has been necessary only for a very small number of vectors in a given velocity data set, if at all (eg., the wake-shear layer flow of Part II had less than 1% of its vectors detected interactively). Later in this section, a preliminary suggestion will be made of a secondary *LSD* filter that may be able to isolate the aforementioned areas of erroneous vectors for replacement, identifying them via their excessively inflated *LSD's*.

The result of the application of the *LSD* method to all vectors in the velocity grid is a list of points sufficiently erroneous to have failed the criterion of equation (20). The potential problem of adjacent bad vectors is then mitigated by *sorting* this list in order of decreasing velocity difference ($|\delta q|$). Thus, a principal advantage of the technique: **the very worst points are the first ones to be replaced.** For example, a vector having a u-/v-component that has $|\delta q| = \Delta$ (where Δ is high enough to fail the *LSD* criterion) will only be replaced *after* another vector with $|\delta q| = \Lambda$, where $\Lambda > \Delta$.

Lower values than $N_{SDF} = 2$ can be applied to a velocity field; this would increase the number of vectors failing the equation (20) criterion. One can consider that in the limit as $N_{SDF} \rightarrow 0$, the number of points to be replaced approaches the total number of grid points, n^2. This case is reminiscent of the spatial low pass filter of Willert & Gharib (1991) referred to earlier (cf. §2.1). The pivotal difference is that this *LSD* "filter" is not an indiscriminate one, ie., one that automatically processes all vectors in the same order every time. One can instead speak of a *sorted* low pass "filter" which replaces

points in order of decreasing error.

Taking N_{SDF} to lower values has the side effect of increasing the amount of time needed to sort an increasingly large list of "bad" vectors. To alleviate this, a twofold or *combination* filter was adopted. The first stage involves analyzing a raw velocity field with the *LSD* filter with a high value of N_{SDF}. This would replace the very worst points, ideally leaving a nearly smoothly varying velocity field. This intermediate vector set can then be processed by a straight low pass filter analogous to that of Willert & Gharib (1991). We expect that the formerly discussed data contamination effects will be small as a result of this filtering combination. Also, as stated earlier, we have greatly reduced — but not totally eliminated — the need for interactive detection and replacement of erroneous vectors.

2.3. Subpixel accuracy

Because of the nature of equations (14) and (16), it becomes necessary to use slightly different methods to determine the accuracy of the simulated and real flows. Especially for real flows, "accuracy" is not necessarily correct language, since it is more difficult to establish a standard for comparison. Measures of statistical uncertainty (Willert & Gharib (1991)) were therefore developed as an adjunct to those of "accuracy." It is implicit that the purpose of the following techniques were to determine the limits of accuracy of our digital-based PIV implementation through testing on simulations. The determination of real flow parameters in §4.1.1 was made in the context of these limits.

2.3.1. Probability functions

In either simulated or real flows, the initial measure of accuracy is derived from the absolute differences in velocity components from equations (14) and (16). These are converted to differences in displacement in units of hundredths of pixels (or pixel hundredths):

$$\begin{aligned} \delta X &= 100 \times |\delta u| \Delta t \\ \delta Y &= 100 \times |\delta v| \Delta t \end{aligned} \tag{21}$$

These displacement differences are then reconstructed as probability *density* functions (or histograms) $p_{\delta X}(\xi)$ and $p_{\delta Y}(\xi)$, in terms of the displacement variable ξ:

$$\begin{aligned} p_{\delta X}(\xi) &\sim \frac{\mathcal{P}\{\xi \le \delta X \le \xi + \Delta\xi\}}{\Delta\xi} \\ p_{\delta Y}(\xi) &\sim \frac{\mathcal{P}\{\xi \le \delta Y \le \xi + \Delta\xi\}}{\Delta\xi} \end{aligned} \tag{22}$$

In the usual sense, these density functions represent the probability that a displacement difference falls between the displacement values ξ and $\xi + \Delta\xi$ (Rosenfeld & Kak (1982)). These functions are calculated over a statistical sample size equal to all velocity grid points on an image (n^2).

Because $p_{\delta X}(\xi)$ and $p_{\delta Y}(\xi)$ are typically very similar functions, either or both will sometimes be denoted $p(\xi)$ for conciseness, especially on plot axes of figures discussed in Part II (the u- and v-component probability density functions are not identified separately on these figures because of this parallelism to one another).

In accordance with the definitions of δX and δY in equation (21), $\Delta\xi$ is taken to be 1 hundredth of a pixel (0.01 pixels). The range of δX and δY is generally set to between 1 and 50 pixel hundredths (0.01 and 0.50 pixels). Any values greater than 50

pixel hundredths — meaning that the measurement of horizontal/vertical distance was more than 0.50 pixels offset from the "proper" value — were simply set to 50 pixel hundredths. This has the effect of manifesting a "kink" in the "tail" of the histogram (the high end). The size of this kink will therefore indicate the probability that a measured displacement difference will have a value greater than or equal to the maximum of 50 pixel hundredths (0.5 pixels). As the initial location of the displacement correlation peak is found to the nearest 1.00 pixels (100 pixel hundredths), it seems reasonable to set a uniform range for the density function to have a maximum of half this amount (cf. §1.4 regarding determination of displacement). There appears to be little need to extend the histogram beyond this point, since any greater value represents an equivalently conspicuous deviation.

Discrete integration of these density functions produce the corresponding probability *distribution* functions,

$$\begin{aligned} P_{\delta X}(\xi) &= \int_0^{\xi} p_{\delta X}(\xi)\, d\xi = \mathcal{P}\{\delta X \leq \xi\} \\ P_{\delta Y}(\xi) &= \int_0^{\xi} p_{\delta Y}(\xi)\, d\xi = \mathcal{P}\{\delta Y \leq \xi\} \end{aligned} \tag{23}$$

The right hand sides of equations (23) are derivable from considerations of calculus and probability theory (Rosenfeld & Kak (1982)). In this sense, $P_{\delta X}(\xi)$ and $P_{\delta Y}(\xi)$ represent the probability that a measurement yielded a displacement less than or equal to ξ. These distribution functions have proven to be more useful in evaluating the accuracy of a velocity field than the density functions.

In an analogous sense to probability density functions, $P_{\delta X}(\xi)$ and $P_{\delta Y}(\xi)$ are

sometimes represented collectively as $P(\xi)$, for the same reasons (again, it has not appeared necessary to distinguish between the closely similar u- and v-component probability distribution functions).

To provide a more concise statement of accuracy, the distribution functions were evaluated at a specific value,

$$P_{\delta X}(\xi) = 0.9 \quad \text{AND} \quad P_{\delta Y}(\xi) = 0.9 \tag{24}$$

These yielded values of ξ which signify that 90% of the vectors of the displacement field were measured to within ξ pixels or less. Note that a displacement field can have 90% of the horizontal displacements (δX) accurate to within one value of ξ, and 90% of the vertical displacements (δY) accurate to a slightly different value of ξ. The conservative, or higher, estimate of ξ was chosen, and designated as $\xi_{90\%}$.*

Thus, a raw velocity field has a single value of $\xi_{90\%}$ associated with it. The effect of filtering with each of the *LSD* and *combination* filters (*LSD* followed by straight low pass filtering) can be seen by comparing the values of $\xi_{90\%}$ before and after filtering. This became a principal criterion for the establishment of the accuracy and limitations of PIV in respect to the various simulations in §3. It is less definitive in relation to real flows, again because of the possible lack of analytical values for the velocity field against which to establish accuracy.

*In an analogous sense, the values for each component could be denoted as $X_{90\%}$ and $Y_{90\%}$.

2.3.2. Statistical uncertainty

Willert & Gharib (1991) reports results concerning the variation of PIV accuracy with particle density and the magnitude of displacement (together with some information regarding sample window sizing). These results were obtained through imaging a pattern of black dots (as particles) on a white background for two cases: several images with no relative displacement; images where particle image pairs were shifted mechanically over a range of possible displacements. After running the images through their PIV system, they apparently determined the *statistical uncertainty* of the velocity field by measuring the RMS fluctuation thereof. "Fluctuation" appears to refer to the variation of results of velocity measurements given multiple image pairs with a common characteristic (ie., several pairs that were mechanically moved the same distance with respect to each other). Both types of experiments were carried out with variation of particle seeding density, ρ_P.

The conclusion reached was that the lowest possible uncertainty was approximately 0.01 pixels, according to their Figure 6b. Upon inspection of that figure, it would seem that this uncertainty corresponds to $\rho_P = 74$ and mechanical displacement of on the order 0.1 pixels. Thus, a fairly high density coupled with a rather small displacement yielded an uncertainty that is 10% of the distance moved. For a more moderate density of $\rho_P = 24$ and the same distance, the uncertainty becomes about 0.02 pixels or 20% of the distance moved. For a larger mechanical displacement — about 1 pixel — with this density, the uncertainty increases to about 0.04 pixels, which is now 4% of the distance moved. It would seem desirable to remain cognizant of the comparative magnitudes of uncertainty versus displacement.

Our own measure of statistical measurement uncertainty is based upon the information obtained from application of the *LSD* method. Consider equation (19), noting the implicit dependence of q_{LSD} upon (n_x,n_y), the particular grid point or vector being examined. The measurement uncertainty of the ***entire*** velocity field can be estimated by averaging the *LSD* field over all values of (n_x,n_y) as follows:

$$\langle q_{LSD} \rangle = \frac{\sum_{j=1}^{n_y} \sum_{i=1}^{n_x} q_{LSD}(i,j)}{n^2} \tag{25}$$

This represents an average of the differences between each vector and its surrounding eight neighbors *for the whole field.* However, wide variations in the function $q_{LSD}(n_x,n_y)$ will not be reflected in $\langle q_{LSD} \rangle$. An indicator of the magnitude of these variations over the *whole* field can be illustrated by the standard deviation from this average $\langle q_{LSD} \rangle$

$$\langle \sigma_{LSD} \rangle_q = \sqrt{\frac{\sum_{j=1}^{n_y} \sum_{i=1}^{n_x} (q_{LSD}(i,j) - \langle q_{LSD} \rangle)^2}{n^2}} \tag{26}$$

Equations (25) and (26) together yield an independent demonstration of the degree of fluctuation of the velocity field. Plots of these uncertainties are included in Part II in addition to those of the aforementioned probability functions.

Further, $\langle \sigma_{LSD} \rangle_q$ presents interesting possibilities for modification of the *LSD* filter. A secondary standard deviation filter can be devised by comparing the individual values of $q_{LSD}(n_x,n_y)$ against $\langle \sigma_{LSD} \rangle_q$ in a vein similar to that of equation (20). This new criterion will generate another list of grid points which have failed it, with the idea being that these erroneous vectors would have excessively inflated *LSD's*. These vectors would then be replaced because of their failure of this secondary *LSD* criterion, even if they had

already passed the primary *LSD* criterion. A preliminary application of this modified *LSD* filter will be briefly discussed in Part III, and is currently under investigation.

2.4. Circulation & Vorticity

During the development of our implementation of PIV, emphasis was placed upon the determination of velocity and velocity error, with the latter establishing some guidelines on the capabilities and limitations of the technique (cf. §3). The importance of velocity-derived quantities should not be underestimated. The ability of PIV to provide velocity data over numerous points simultaneously is certainly advantageous: direct numerical integration and/or differentiation of the velocity field is sufficient to produce desired quantities such as vorticity and circulation.

The standard Cartesian formulations used in our technique will be outlined in §2.4.1. Since the simulations of Part II are based on a velocity field (solid body rotation) that is more easily described in polar coordinates, an alternative set of formulas in §2.4.2 was developed for that case to provide an estimation of the accuracy of PIV.

2.4.1. Cartesian formulation

Since the obtained velocity data is Cartesian by definition, this section represents the default formulation. The circulation was obtained by calculating the line integral of velocity around a four-point square contour,

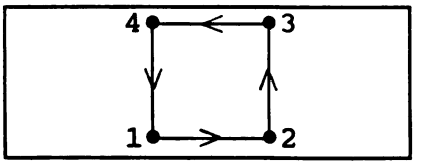


Figure 6. $\Gamma(x,y)$ integration contour.

$$\Gamma(x,y) = \oint \vec{u}(x,y) \cdot d\vec{l} \qquad (27)$$

The discrete integration is performed by assuming linear behavior in the velocity over a small interval.* The approximation is simply,

$$\begin{aligned} \bar{q}_1 &= \frac{u_1 + u_2}{2} \quad , \quad \bar{q}_2 = \frac{v_2 + v_3}{2} \\ \bar{q}_3 &= \frac{u_3 + u_4}{2} \quad , \quad \bar{q}_4 = \frac{v_4 + v_1}{2} \end{aligned} \tag{28}$$

Since the velocity grid is square, all distances between nodes are equal (the possibility that future implementations may allow for rectangular node spacing has been taken into account). The expression for the circulation $\Gamma(x,y)$ becomes

$$\Gamma(x,y) = \sum_{i=1}^{4} \bar{q}_i \cdot (\Delta l)_i$$

$$\textit{where } (\Delta l)_i = \left\{ \begin{matrix} +\Delta l \ , & i = 1,2 \\ -\Delta l \ , & i = 3,4 \end{matrix} \right\} \tag{29}$$

The *area-averaged vorticity* is then available directly from the previous equation through division of the appropriate area (ie., the area bounded by the contour of Figure 6):

$$\omega(x,y) = \frac{\Gamma(x,y)}{(\Delta l)^2} \tag{30}$$

This vorticity definition is sometimes referred to as the *microcirculation.*

A differential version of vorticity will also be instituted as an option. The formula would be a first order approximation based on finite differences involving the two nodes on either side of the center node, in each dimension. A limitation on this variant is that the approximation to the first order derivatives may be invalid if the node spacing is too large. Naturally, this is more likely to be true for a non-overlapping sample window grid. By contrast, Willert & Gharib (1991) use a second order

*Analogous to the trapezoidal integration approximation, in two dimensions.

approximation with their 75% overlapping grid. Nevertheless, even an overlapping grid should be designed with care to ensure that the node spacing — and therefore the length scale of the derivatives — is appropriate for the scales of motion of the flow being investigated.

2.4.2. Radial formulation

Circulation and vorticity

The calculation of circulation and vorticity were used to confirm the accuracy of the PIV technique when tested upon a simulated solid body rotation (*SBR*) flow field. As stated previously, a polar formulation was used for better comparison with the theoretical circulation for this flow, which can be derived from the *SBR* velocity field,

$$\begin{aligned} u_r &= 0 & u &= -\Omega y \\ u_\theta &= \Omega r & v &= \Omega x \end{aligned} \tag{31}$$

where Ω is the angular rotation rate. Substitution of the above into equation (27) and integrating around an appropriate contour yields,

$$\Gamma(r) = \frac{\Omega}{2}\pi r^2 \tag{32}$$

for a quarter-circle of radius r (for simplicity, the actual *SBR* flow field used was a solid "quarter-body" rotation flow). The new circulation function $\Gamma(r)$ was initially calculated by two different methods: 1) surface integration of vorticity over a quarter-circular area; 2) circular line integration of velocity.* A comparison showed the latter algorithm to be more accurate in our case. It therefore became the method of choice, and is the

*Strictly speaking, the radial circulation function is actually calculated using (n_x,n_y) grid coordinates, then converted to pixel coordinates (cf. Appendix §A.2 for more on coordinate systems). Consequently, $\Gamma(n_r)$ would actually be more appropriate notation, but $\Gamma(r)$ is used for brevity. For example, Figure 7 uses grid coordinates.

subject of discussion for the rest of this section.

As just expressed, the simulation encompasses a single quadrant of a hypothetical circular body. The integration therefore proceeds around successively increasing quarter-circular contours, which can be separated into three parts,

$$\Gamma(r) = \Gamma_x(r) + \Gamma_y(r) + \Gamma_c(r) \tag{33}$$

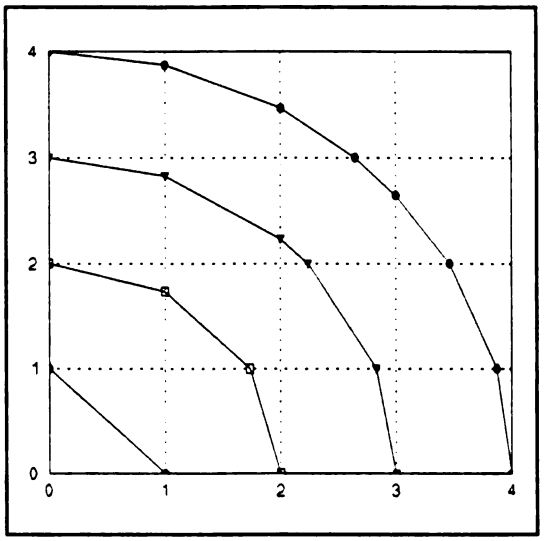


Figure 7. Subdivision of contour arc sections into straight line segments.

with Γ_x and Γ_y are the horizontal and vertical portions of the contour, and Γ_c comprises the curved section. Figure 7 shows some typical examples of the segmentation of curved portions of the contours. Details of evaluation of the respective terms may be found in Appendix C. The basic premise is to calculate average u-/v-components over each segment and perform a discrete summation over all the segments (analogous to equations (28) and (29)).

The area-averaged vorticity (or microcirculation) is again directly derivable from circulation. The appropriate area for a given $\Gamma(r)$ is of course that of a quarter circle of radius r, leading to the following expression for $\omega(r)$,

$$\omega(r) = \frac{\Gamma(r)}{\frac{1}{4}\pi r^2} \tag{34}$$

It should be emphasized that in our case that ω is evaluated at r instead of at the centroid of the area. Comparison of the results of equations (33) and (34) for the simulations versus the theoretical *SBR* values are contained in §3.

Circulation error

Two indicators were used to determine the circulation error of the simulation measurements. The first compares the measured value to the theoretical value at a given radius (ie., error relative to local circulation) as follows,

$$E_{LOCAL} = \left| \frac{\Gamma_{THEO} - \Gamma_{MEAS}}{\Gamma_{THEO}} \right| \qquad (35)$$

To provide a standard for comparison, an exact theoretical velocity field was constructed and evaluated at coordinates corresponding to the same sample window grid as that of the simulations (this, then, would represent an application of the PIV technique yielding perfect results with zero error).

The nature of the equation (35) method is to artificially inflate errors at low radii. This is where the true circulation is low, magnifying discrepancies between true and measured values, because equation (35) divides small numbers into other small numbers. The measured circulation may actually be quite close to the theoretical value, but the *fractional* discrepancy may be large.

This problem is mitigated by the second method for obtaining circulation error, which utilizes the identical difference between theoretical and measured values at each radius and normalizes this difference with the aforementioned maximum circulation rather than the theoretical local circulation (ie., error relative to maximum circulation):

$$E_{MAX} = \left| \frac{\Gamma_{THEO} - \Gamma_{MEAS}}{\Gamma_{MAX}} \right| \qquad (36)$$

Γ_{MAX} evaluated at $r \approx 500\sqrt{2}$ pixels

The value of r was chosen to be of equal magnitude to the diagonal dimension of a

512×512 pixel^2 image within conditions of definition of the sample window grid. Note that this method does not artificially inflate errors at low radii. Equations (35) and (36) form the basis of the simulation circulation error results in §3.

Note on integration schemes

The integration techniques discussed in §2.4 are simple extensions of the basic trapezoidal rule. It is certainly possible to improve these routines, perhaps by resorting to Simpson's rule or a type of quadratures. This is a topic for future experimentation, and is further addressed in Part III. Nevertheless, the Part II results using the current integration schemes appear to generate well-behaved functions without introduction of significant error.

Part II. TESTING & APPLICATION

3. SIMULATIONS

Background

Extensive simulations were performed during the development of this PIV implementation. Their purpose was to define the limits of the operating regime of the technique. It became apparent that certain parameters — most notably particle density — had to be constrained to specific value ranges. These are discussed in §3.2-3.3, and placed in the context of Keane & Adrian (1990), Willert & Gharib (1991), and Lourenco & Krothapalli (1987) when appropriate.

Definition of the PIV operating envelope by these parameters then led to application of the technique to real flows. The focus of experimentation became to process real flow images using the criteria established by the simulation results.

3.1. Artificial image generation

The solid body rotation flow field was selected because it offers a large dynamic range in velocity magnitude and direction along with uniform vorticity. The velocity field in both polar and rectangular coordinates is given by equation (31), rewritten below:

$$\begin{aligned} u_r &= 0 & u &= -\Omega y \\ u_\theta &= \Omega r & v &= \Omega x \end{aligned} \tag{37}$$

512 × 512 pixel^2 images were generated to reflect this velocity field. These images were randomly seeded in space with individual particles having a two-dimensional Gaussian intensity profile. These intensities were chosen to conform with the range

given by 8-bit images — those that have 256 shades of grey (or a 0-255 grey scale). The actual Gaussian particle intensity distribution is as follows:

$$I = I_0 \exp\left[-\frac{(x^2+y^2)}{\sigma^2}\right] \qquad (38)$$

$$\textit{where:}\quad I_0 = 127,\quad \sigma = 0.95,\quad x,y \in [-2,2] \text{ pixels}$$

The Gaussian peak value and scaling factor were chosen to yield "diamond-shaped" particles with major axes measuring 5 pixels:

$$I(x,y) = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 \\ 0 & 14 & 42 & 14 & 0 \\ 2 & 42 & 127 & 42 & 2 \\ 0 & 14 & 42 & 14 & 0 \\ 0 & 0 & 2 & 0 & 0 \end{bmatrix} \qquad (39)$$

Figure 8 shows an example of an artificial image seeded with more than 6000 particles corresponding to equation (39).

A second image was derived from the first artificial image by translating each particle through a distance indicated by the local velocity vector and the given time interval Δt between images. In these simulations, the angular rotation rate Ω is set to unity with a non-dimensional time increment of $\Delta t = 0.01$ (one can consider Δt to have units of seconds — which yields an image acquisition frequency $1/\Delta t = 100$ *Hz* — without loss of consistency). This yielded a particle displacement range varying from 0 to 5 pixels.

Having created an image pair, the next step was to apply the PIV technique by

using the main velocity processor of §1. The resulting "raw" velocity field was processed by the *combination* filter, which consists of the *LSD* filter followed by a straight low pass filter (cf. §2.2.2). For all of the following, $N_{SDF} = 2$ was used to set the selectivity of the *LSD* filter. The subpixel accuracy is determined before and after the filtering process (cf. §2.3).

The first of the following sections establishes the importance of particle density, ρ_P. In this section, explicit reference will be made to the effects of each stage of the aforementioned filtering process upon subpixel accuracy. The later sections will only discuss the results after the filtering process. Figures with intermediate results will nevertheless be mentioned briefly.

3.2. Variation of particle density

Images corresponding to four different densities were chosen. These densities were arbitrarily chosen based on a 64 × 64 pixel^2 area. The selected values were ρ_{64} = 192, 96, 72, and 48 particles per the foregoing area. The total number of particles within a 512 × 512 pixel^2 image can be obtained by multiplying the previous figures by 64 (for example, Figure 8 has $\rho_{64} = 96$, which corresponds to 6144 particles). Thus, four image pairs corresponding to these values were generated.

Velocity fields

Prior to use of the velocity field processor, the sizes of the sample and roam windows, S and R were chosen (cf. §1.2). This choice is dictated by the magnitude of the maximum pixel displacement being measured, which is 5 pixels, which implies that

$$\Delta_{MAX} = \frac{R - S}{2} \geq 5 \text{ pixels} \qquad (40)$$

The parameter Δ_{MAX} denotes the maximum resolvable displacement for a particular (R,S) pair.

The lowest value of S that satisfies equations (40) and (3) is $S = 15$ pixels, which in turn leads to $R = 25$ pixels.* In practice, using this combination sometimes results in truncated correlation peaks if a displacement of about 5 pixels is being measured. Therefore, the equality in equation (40) is usually disregarded. In most cases, the measured displacement is simply increased by 1 or 2 pixels for a working value of Δ_{MAX}. In this case, $\Delta_{MAX} = 6$ pixels was chosen, leading to $(R,S) = (27,15)$. This leads to a 33 × 33 nonoverlapping grid, or 1089 velocity vectors (by equation (2)).

It has appeared evident that previous applications of PIV by other researchers defines the particle density ρ_P as the number of particles per sample or interrogation window (cf. Introduction). Given our simulation's sample size, ρ_P can now be defined relative to the 15 × 15 pixel^2 sample window. The values corresponding to those of ρ_{64} = 192, 96, 72, and 48 are given by $\rho_P = 10.55, 5.27, 3.96,$ *and* 2.64 respectively (these are the actual values to be used in the rest of §3).

By comparison, Willert & Gharib (1991) reported having $6 \leq \rho_P \leq 74$ in their simulations and $10 < \rho_P < 20$ in their real flows. The latter density range is exactly the *minimum* density range recommended by Keane & Adrian (1990), although there is some question as to whether their criterion should be halved because it is made in the

*Equation (3) represents the guideline used by Willert & Gharib (1991), and was used to delimit S and R to enable an analogous simulation. This guideline can also be a starting point for window sizing for real flow analysis by our PIV system. Usually, the values of Δ_{MAX} and ρ_P are more central to the eventual choices of S and R.

context of a double-pulsed PIV system. Keane & Adrian (1990) use the terminology, *"image density,"* which possibly implies that it refers to particle images, not particles (a double-pulsed system produces approximately twice as many images as particles). Lourenco & Krothapalli (1987) suggests that $\rho_P \geq 4$ is sufficient. Later in this section, it will be shown that our simulation results agree with this suggestion.

The "raw" velocity fields generated for this (R,S) pair and these densities are in Figure 9, followed by the intermediate and final velocity fields produced by the filtering process (in Figure 10 and Figure 11). The run-time for production the raw velocity field with this (R,S) combination was approximately *7 minutes* on a 25 MHz 80386-based personal computer with coprocessor. By contrast, filtering is a fairly quick process: on the order of 1 minute.

Diagnostic information from the performance of the velocity field processor upon this grid was initially used to test and debug the software. The results of a particularly useful diagnostic are shown in Figure 13, a histogram of the number of particles per sample window that were *actually found* in the corresponding roam window (ie., number of particle "matches").* The figure was generated through analysis of all 1089 velocity vectors: this is the total number of occurrences. For example, there were 99 sample windows where the number of particle matches was between 4 and 5. Alternatively expressed, between 4 and 5 particles per sample window were actually "found" in the corresponding roam window, for 99 sample windows out of 1089.

Examination of this histogram, including an analysis of the most erroneous

---

*This was estimated by dividing the correlation peak magnitude by the square of the particle peak intensity I_0 (cf. equations (38) and (39))

vectors shown in Figure 14, showed that most of the worst vectors were produced by sample windows with less than 3 particle matches. Those with 3-4 particle matches seemed to be on the border of tolerance, with more than half resulting in acceptable vectors (this was ascertained through examination of the actual vectors). Most sample windows containing 4 or more particle matches appeared to result in satisfactory vectors. This was the initial evidence that $\rho_p = 4$ seemed to be an acceptable minimum, which agrees with Lourenco & Krothapalli (1987) as expressed earlier.

Consider again the three sets of velocity fields. Note that even the raw, unprocessed velocity fields of Figure 9 show most of the vectors as being measured satisfactorily. Note the trend of increasing numbers of erroneous vectors as a function of decreasing values of ρ_p. Next, Figure 10 represents the intermediate step in velocity post-processing: after the *LSD* filter but before the straight low pass filter. Since *LSD* filter only replaces the worst points, one observes that most of the vectors are unchanged. Yet the improvement of the data (*without* interactive selection of points) is obvious. Finally, Figure 11 displays the final velocity fields, after the complete filtering procedure. Close inspection reveals minor differences between the vectors of Figure 10 and Figure 11. Careful scrutiny also still reveals deviations in vector orientation and magnitude as a result of the lowering of particle density.

Velocity-derived quantities

During the development of our PIV implementation, the main emphasis has been placed on determination of velocity. Nevertheless, preliminary methods for calculation of circulation and vorticity were developed. These methods are detailed in §2.4.

Particular velocity data sets were processed for determination of cartesian and especially *radial* circulation and vorticity ($\Gamma(r)$ and $\omega(r)$). While the cartesian formulation would be preferred for many flows, the radial formulation better enabled the confirmation of the efficacy of both the PIV software and the circulation/vorticity algorithms, due to the inherent radial nature of the solid body rotation (*SBR*) flow field.

This testing of the PIV implementation involved the use of an artificially generated *theoretical* velocity field, evaluated at the same pixel locations using the identical sample window grid (this is the same standard velocity field used for calculation of circulation error in §2.4.2). Figure 12 displays the four radial properties discussed in §2.4.2 as functions of radius for a single value $\rho_P = 5.27$. The two properties besides circulation $\Gamma(r)$ and vorticity $\omega(r)$ are the two indicators of circulation error: E_{LOCAL} and E_{MAX}. The "plus" symbols on parts (a) through (d) represent measured results corresponding to a filtered version of a $\rho_P = 5.27$ velocity field. The solid lines on all plots correspond to the results of the aforementioned theoretical velocity field, which can be considered to represent the *best possible* performance that could have been expected of PIV.

Part (a) of Figure 12 shows the circulation normalized by a predefined maximum circulation, Γ_{MAX} (again, cf. §2.4.2). The solid theoretical line follows the expected parabolic shape of equation (32), and the measured circulation shows very good agreement with this parabola.

A characteristic of solid body rotation is that of constant vorticity throughout the flow, as shown by the horizontal line corresponding to the theoretical velocity field. The value of this line is $\omega = 2$ because the angular rotation rate Ω was set to unity. The

measured vorticity appears a bit erratic at low radii before smoothing out at higher values of r. This is a result of the means by which the circulation contour is defined (recall that vorticity is derived by division of circulation by area in this case, using equation (34)). The integration contour used is that of a quarter-circle, with the curved portion being simulated by a series of line segments. Through consideration of Figure 7, it is immediately perceivable that — at low radii — there are not sufficient line segments to adequately simulate a circular contour. This approximation is therefore not very good here, resulting in the observed aberrational values of vorticity at low r. Inspection of $\omega(x,y)$ (not shown here) showed that vorticity was actually well behaved in these regions.

The two indicators of circulation error are shown in parts (c) and (d) of Figure 12. It is apparent that the measurement error in both cases is fairly low, with an increase at either low or high radii. For part (c), the error E_{LOCAL} is higher at low radii because it is based on division by the local theoretical circulation (cf. equation (35)). At low radii, both measured and theoretical quantities are small, and division of one small number by another could result in a relatively large number (also see page 45).

The slight increase in error E_{MAX} of part (d) at high radii has a different cause. In this region of the flow, the displacements are the highest in both horizontal and vertical directions. As a result, the process of 2-D cross-correlation results in a correlation peak near the edge of the correlation domain. If this correlation peak is partially truncated (or degraded in some other sense), then the 2-D Gaussian fitting processing will be less reliable. As circulation is based on the calculation of the line integral of velocity around a contour, these errors will affect to the integration process. Nevertheless, note that the maximum of the ordinate axis of part (d) is merely 4%, and

even the largest point of error at high radii has E_{MAX} at only about 1%. Therefore, this error source does not appear to be significant.

Although the effects of variation of particle density ρ_P could be recognized from comparing their respective $\Gamma(r)$ and $\omega(r)$, it was evident that analysis of *displacement error* probability density and distribution functions was the appropriate choice for discerning a minimum value for ρ_P. The entire premise of establishing such a criterion is based on improving *velocity* measurement using PIV. This was also the case with the analysis of §3.3 in determining a limit for out-of-plane motion. Furthermore, the emphasis was placed upon improvement of velocity measurement in the wake-shear layer flow of §4. Therefore, circulation and vorticity results will not be shown in the rest of Part II. Nonetheless, some suggestions for enhancing the algorithms for deriving these quantities will be discussed in Part III.

Probability density functions (normalized histograms)

Effects of variable density are more readily discernable upon viewing the velocity component probability density and distribution functions (the latter will prove to be best for this purpose). These are based upon the differences between the theoretical (equation (37)) and measured velocity fields (cf. §2.3.1 for definitions of probability functions).

The probability density functions, $p_{\delta X}(\xi)$ and $p_{\delta Y}(\xi)$ are displayed for each value of ρ_P in Figure 15, with two curves each for the raw, *LSD*, and final velocity fields. These represent the u- and v-component functions considered separately. Observe that these curves are very close to one another (thus, there does not appear to be a need to distinguish them). For example, for the $\rho_P = 5.27$ raw velocity field, about 3-4% of the

1089 velocity nodes are seen to have measured displacements that differed from the predicted theoretical displacements by about $\xi = .10$ pixels.

Note further that there does not appear to be a significant change between the raw and *LSD* velocity field density functions. Recall again that the *LSD* filter only replaces a portion of the data. This is a very significant portion, however, in that it generally consists of those vectors that have displacement differences higher than 0.5 pixels. It is possible to recognize that the final velocity fields have greatly reduced "kinks" in the "tails" of the density functions; this becomes more discernible upon viewing the plots of the distribution functions.

The near elimination of all such higher displacement difference vectors by the *LSD* filter leads to a significant shift towards lower displacement differences (ie., higher accuracy) from the previous velocity fields to the final velocity field. Because a very large proportion of the velocity nodes were measured with fairly high accuracy, a subsequent process of indiscriminate low-pass filtering apparently spread the "influence" of these points as the velocity data was being smoothed.

Probability distribution functions

Most of the previous phenomena is more evident upon consideration of the probability distribution functions, $P_{\delta X}(\xi)$ and $P_{\delta Y}(\xi)$ in Figure 16. Once again, there are u- and v-component curves for the raw, *LSD*, and final velocity fields. For these functions, the differences between the three stages of velocity fields are clearer. Observe that the difference between raw and *LSD* velocity fields increases as a function of decreasing ρ_P. At $\rho_P = 10.55$, there are so many particles per sample that only a few

points were so bad as to be replaced by the *LSD* filter. At $\rho_p = 2.64$ (which is less than the previously suggested minimum of $\rho_p = 4$), a larger portion of the velocity field was replaced.

Note the substantial improvement in accuracy as a result of this filter, despite the largely insufficient particle density. Even more dramatic improvement is recognizable in the final velocity fields. For example, in the $\rho_p = 5.27$ case, 90% of the velocity vectors were measured to within about $\xi = .06$ pixels in displacement. For comparison, Fincham et al (1991) asserts that "particles can be located to within .05 pixels under perfect conditions ..." for their implementation of DPIV.

The ease of discerning differences between $P(\xi)$ curves as compared to $p(\xi)$ curves made the former preferred for analysis (recall that the lack of subscripts in the notation $P(\xi)$ denotes either or both component distribution functions, with equivalent meaning for $p(\xi)$).

Velocity field accuracy

The previous $P(\xi)$ example — that 90% of the vectors were measured to within $\xi = .06$ pixels for $\rho_p = 5.27$ — may be described alternatively as $\xi_{90\%} = .06$ pixels (cf. 2.3.1 for definition of $\xi_{90\%}$). Consider the meaning of this subpixel accuracy value: for a velocity field with a displacement dynamic range of 0 to 5 pixels, 90% of the velocity field was measured to better than or equal to .06 pixels. This value of $\xi_{90\%}$ is thus about 1% of the maximum displacement. This has proven to be a useful parameter for summarizing the results of a probability distribution function.

Figure 17 shows two plots: the right side plot displays $\xi_{90\%}$ versus ρ_p; and the

left side plot has the percentage of the 1089 velocity grid points failing the *LSD* criterion — F_{LSD} — shown versus ρ_P. Considering the right side plot, the $\xi_{90\%}$ curve corresponding to the raw velocity field is the highest curve. From highest to lowest particle densities, the range is from $.14 \leq \xi_{90\%} \leq .43$ pixels. The *LSD* curve shows greater improvement at lower densities, which agrees with the observations on $P(\xi)$. The final curve shows nearly uniform improvement at all densities and its value is nearly fixed at $\xi_{90\%} \approx .06$ pixels for all values of ρ_P. This means that the final filter had a nearly equal smoothing effect on the velocity data regardless of the density.

Now consider the F_{LSD} curve in the same figure. Only about 3% of the velocity grid failed the *LSD* criterion at $\rho_P = 10.55$. This value increases with decreasing density until it reaches $F_{LSD} \approx 17\%$ at $\rho_P = 2.64$. Thus, 17% of the velocity field was replaced in this instance, which is not unsatisfactory considering the low particle density. The minimum $\rho_P = 4$ criterion again appears sufficient in this context, since the change in ρ_P from 10.55 to 3.96 increased the value of F_{LSD} to only 7%.

An alternative method of examining the information from the $P(\xi)$ plots is to construct lines of constant accuracy as a function of ρ_P (these will be termed "iso-accuracy" plots). The plots for all three velocity field filtering steps are included in Figures 18-20. There are u- and v-component plots for each of the raw, *LSD*, and final stages of the velocity field, with the y-axis scaling decreasing with each stage. Note that the curves marked 90% in each of Figures 18-20 are identical to those displayed in Figure 17. The curves corresponding to 40% and 50% are also notable in all three velocity stages. This is especially true in the final velocity stage, which shows that 50% of the velocity grid is accurate to within an amazingly low 0.02 pixels (except for $\rho_P =$

2.64 in the v-component plot). This confirms the previous hypothesis regarding the shifting of probability density functions: a substantial proportion of the vectors have been measured to a fairly high accuracy.

Default particle density

Inspection of these iso-accuracy plots once again shows that most of the accuracy losses occurred at $\rho_P = 2.64$, while $\rho_P = 3.96$ seemed to perform nearly as well as higher values of ρ_P. Together with the previous data, it seemed appropriate to accept $\rho_P = 4$ as a minimum. It was realized, however, that the forthcoming studies of three-dimensional effects would involve deleterious effects on "effective" particle density within a sample window. Thus, $\rho_P = 5.27$ was chosen as the default density. This velocity data set would have a level of tolerance higher than the minimum density, but would still display analyzable error levels if the out-of-plane motion was sufficiently large. The highest density was not considered because some preliminary experimentation with real flows revealed that 10 particles per window was not a common circumstance. Recall that we do not seed the flow with extraneous particles as a part of our PIV implementation, but depend upon the naturally occurring dirt particulates in the flow.

3.3. Out-of-plane motion

Two different versions of three-dimensional losses were analyzed: random dropout of particles from the flow, with simultaneous one-for-one addition (§3.3.1); and w-components indicated by random fluctuation of particle intensity (§3.3.2). The intent of the former was to suggest a tolerance level for the fraction of particles moved, F_ρ, out

of the viewing plane as a result of 3-D motion. This is not the same as specification of a w-component limit, which Keane & Adrian (1990) does; however the two criteria may well be related for some flows. The purpose of the intensity variation study was to supplement the previous one with a correlation between a limiting w-component and the velocity accuracy (as discussed in §2.3 and §3.2).

The results that the following sections will detail are: a maximum value of F_ρ = *20-25%* is suggested; however, there was no explicit correlation between intensity-based w-component and velocity accuracy, although a general decrease in accuracy is noted. It is interesting that Keane and Adrian (1990) suggest a similar figure of *25%* for a w-component threshold, based on their own simulations. It is therefore conceivable that the limitations on F_ρ and w-component may be related in some sense.

3.3.1. Effect of removal/addition of particles

In this investigation, a 3-D motion was superimposed upon the 2-D solid body rotation flow field. During the *SBR* motion: 1) a fraction (F_ρ) of the total number of particles is *randomly removed* from the image; 2) an *equal* fraction is *replaced in different locations* onto the image. Thus, the overall particle density is maintained at the same value. The end result is a sudden, total disappearance of a number of particles between the first and second flow images, together with an equally precipitous addition of completely new particles during the same time interval.

Since PIV attempts to resolve velocity vectors through correlation of particles between consecutive images, each of the preceding phenomena causes errors. Each particle removed per sample causes a reduction in the displacement correlation peak by

one "particle match." Each particle added to a sample causes spurious correlation peaks, since existing particles will match with the additional one to produce displacement information which has nothing to do with the actual flow field. A sufficient number of such "particle mismatches," combined with the loss of previous "matches," would produce a false displacement peak. This would occur through both reduction of the true peak and enlargement of false ones until a false peak is higher than the actual one.

The crux of the study was to determine just what value of F_ρ was sufficient to effect the above losses to a large degree. Five values of F_ρ were chosen: *5%, 10%, 25%, 40%,* and *50%*. The source images with the default $\rho_P = 5.27$ were selected as the base images from which the five additional image pairs were created. The base images will be considered as the $F_\rho = 0\%$ case. For example, in the $F_\rho = 25\%$ image pair, the average density is still 5.27 particles per sample, but an average of 1.32 particles were randomly removed, with the same average amount added in different locations.

Velocity fields

The raw and *LSD* velocity fields for all cases of F_ρ are shown in Figure 21 and Figure 22. As stated previously, the main focus will be on the final velocity fields here: these are shown in Figure 23 (although some additional information on the intermediate stages will be provided). Figure 23(a) is the base velocity field, and is identical to Figure 11(b). It can be observed that the $F_\rho = 5\%$ and $F_\rho = 10\%$ cases do not show conspicuous deviation from the base data. Alternatively expressed, 3-D motion affecting only 10% or less of the particles does not degrade the accuracy significantly.

Conversely, the $F_\rho = 40\%$ and $F_\rho = 50\%$ cases show a great deal of anomalous behavior. The latter especially displays a wide variation in vectors which sometimes disagrees with the true flow pattern over a somewhat large regions. The $F_\rho = 25\%$ is the most noteworthy velocity field here. Initially, it doesn't appear to differ overmuch from the lower F_ρ cases. Upon further scrutiny, however, slight irregularities can be observed.

To reconfirm the findings of this and other cases — along with the nondependence on a particular simulation — three additional sets of source images were generated and processed (in fact, the $p(\xi)$ and $P(\xi)$ functions for these simulations are based on averages over these four image data sets). Each of the image sets shows similar hints of aberration in the $F_\rho = 25\%$ case. Thus, we arrived at the limiting tolerance of F_ρ was possibly slightly lower than 25%, but probably not lower than 20%. That is, a given flow would be able to tolerate 3-D motion affecting perhaps 20-25% of its particles. This premise is supported by the analysis of the next section.

Probability distribution functions

As stated previously, the probability density functions, $p(\xi)$, are less instructive than the distribution functions (they are nonetheless plotted in Figure 24 for all three velocity stages). Figure 25 shows the probability distribution functions for the six values of F_ρ for the raw, *LSD*, and final velocity fields. Each curve represents an average over eight different curves: one $P_{\delta X}(\xi)$ and one $P_{\delta Y}(\xi)$ curve plotted for each of the four velocity fields (trials) corresponding to the same value of F_ρ. Substantial collapse (over the four trials, not shown here) of these functions was observed — especially in

Figure 25(a) which corresponds to the base velocity field. The subpixel accuracy therefore appeared quite consistent for the same ρ_P over the different trials. For example, $\xi_{90\%} = .06$ pixels was noted for all eight u- and v-component curves for the final velocity fields (this is the same value as for $\rho_P = 5.27$ in §3.2).

Parts (b) and (c) also evidenced good collapse onto a narrow functional region (again, this is not shown here). As will be shown later, $\xi_{90\%}$ increases slightly for these cases. Parts (e) and (f) are horrendous, indicating that 3-D motions affecting a substantial portion of particles in a flow have a strongly adverse effect upon accuracy (this naturally confirms what is expected from intuition).* Note the especially bizarre behavior of part (f) of Figure 25, which shows that the final velocity field is less accurate than the raw velocity field. This clearly indicates that the combination filtering process has been compromised. Because the curves for different trials are sufficiently similar, later figures will also plot the average values of the curves (some will have error bars indicating the minimum and maximum values measured from the four trials).

Part (d) of Figure 25 corresponds to $F_\rho = 25\%$, which is again the most noteworthy 3-D motion fraction. Observe that there exists a definite loss of accuracy here, but the loss is not so excessive as to indicate unreliability in the filtering process as displayed by parts (e) and (f). There is still evidence of substantial improvement from the raw velocity field to the final one; however, there remains a "kink" in the "tail" of the distribution, indicating that a small percentage of the vectors were not quite accurate to within 0.5 pixels. From this fact, it was suspected that a value somewhat less than

*However, even these curves did not show great variations over the four trials.

$F_\rho = 25\%$ might represent the true limiting tolerance for a flow being investigated.

Velocity field accuracy

The above F_ρ criterion is further corroborated by Figure 26, which displays $\xi_{90\%}$ as a function of F_ρ for the raw, *LSD*, and final velocity fields (the right side plot). The centered symbols represent the functional values averaged over the four trials. The extent of the error bars indicates the maximum and minimum functional values obtained (ie., each from a particular trial). For example, the F_{LSD} vs. F_ρ (left side) plot shows that an average (over four trials) of slightly less than 6% of the velocity field failed the *LSD* criterion for $F_\rho = 0\%$. One particular trial had F_{LSD} below 5% (the minimum); another had F_{LSD} at nearly 7% (the maximum). In the $\xi_{90\%}$ vs. F_ρ plot, observe that the error bars do not extend beyond the size of the centered symbol for $F_\rho \leq 10\%$, indicating little differences between the four trials.

Examination of the $\xi_{90\%}$ vs. F_ρ plot displays the raw velocity field showing a sizable loss of accuracy from $F_\rho = 10\%$ to $F_\rho = 25\%$ and tremendous further losses for higher F_ρ values. It is useful to recall that the maximum flow displacement in one dimension is 5 pixels.* Indeed, the degradation of the velocity data occurs to such an extent for the higher F_ρ*'s* that even the *combination* filtering process may no longer be reliable: the final velocity field of the $F_\rho = 40\%$ case is as erroneous as that of the $F_\rho = 50\%$ case.

---

*The $F_\rho = 50\%$ case for the raw velocity field is rendered ludicrous in this context. Even the final velocity field can only suggest accuracy of 1.5 pixels, which is 30% of the maximum displacement. The entire premise of digital PIV is to obtain *subpixel* accuracy.

The filtering of the $F_\rho = 25\%$ case shows a useful result: $\xi_{90\%}$ for the *LSD* and final velocity fields is actually quite close to those of lesser F_ρ. This implies that the vast majority of vectors in the raw velocity field actually had reasonable accuracy; yet enough remained such that 90% of the points were only accurate to a relatively high figure. This is borne out by the u- and v-component "iso-accuracy" plots of Figure 27, which are for the final velocity field, with each curve representing a constant value of ξ averaged over the four trials. For $F_\rho \leq 25\%$ all curves up to $\xi_{80\%} \approx 0.10$ pixels are closely packed. The $F_\rho = 0\%$ case, for example, shows that 40% of the velocity vectors were measured to within about *0.02* pixels, whereas 80% of the vectors were accurate to within roughly *0.05* pixels. The $F_\rho = 25\%$ is only slightly worse, with $\xi_{40\%} \approx 0.03$ pixels and $\xi_{80\%} \approx 0.10$ pixels. The $\xi_{90\%}$ curves show more significant separation from the other curves, but still remain fairly low for $F_\rho \leq 25\%$.

The F_{LSD} vs. F_ρ graph on Figure 26 shows a steadily increasing fraction of points necessitating replacement by the *LSD* filter. It should be realized that this process was rendered moot for high F_ρ: even though 18-20% of the vectors were replaced in this fashion, the remaining data were still somewhat erroneous, making the subsequent straight low pass filtering process almost meaningless. Still, for lower F_ρ, the increase of F_{LSD} as F_ρ increased nevertheless yielded nearly equivalent accuracy. Collectively, these again signify a level of tolerance for F_ρ: values below the tolerance give nearly equal accuracy; values about it are no longer reliably filterable.

Statistical uncertainty

The previous conclusions are also supported by the measurements of statistical uncertainty conducted according to equations (25) and (26) in §2.3.2. These are compared to explicit uncertainty information provided in Willert & Gharib (1991) (see §2.3.2 for a brief description of their methodology). They obtained "RMS fluctuation" values for both simulations and actual experiments. Since the latter was given as a single figure, it is plotted along with our own statistical data in Figure 28 and Figure 29.

Figure 28 shows the variation of displacement measurement uncertainty with F_ρ, with each plot showing the u- and v-component data respectively. The data have been accumulated from all four trials for each F_ρ: the lowest and highest values correspond to the ends of the error bar, and the symbol actually represents the average over the four trials. The dashed curves correspond to $\langle q_{LSD} \rangle$ (the average velocity *LSD* over the entire field), and the solid curves correspond to $\langle \sigma_{LSD} \rangle_q$ (the standard deviation of the average velocity *LSD* over the entire field). As an average of the local standard deviations, $\langle q_{LSD} \rangle$ is an indicator of the *uncertainty of measurement* of displacement over the entire velocity field: this means that a given value of $\langle q_{LSD} \rangle$ for a velocity field indicates that a particular velocity vector will have an average fluctuation (or *LSD*) of $\langle q_{LSD} \rangle$ pixels. Further, the standard deviation of *this* fluctuation is represented by $\langle \sigma_{LSD} \rangle_q$.

From either the u- or v-component plot, it can be seen the average uncertainty $\langle q_{LSD} \rangle$ of the $F_\rho = 0\%$ case is about 0.025 pixels, with the uncertainty standard deviation $\langle \sigma_{LSD} \rangle_q$ having the same numerical value. Thus, the degree of fluctuation of a particular measured velocity vector is 0.025 pixels (again with the same numerical value for the uncertainty of *that* fluctuation). Note that this is not the same type of fluctuation as that

of Willert & Gharib (1991), whose fluctuation measurements are not necessarily based on whole velocity field. For reference, the displayed solid symbols represent the uncertainty values from the vortex ring experiment of Willert & Gharib (1991).

Observe that the average fluctuation, $\langle q_{LSD} \rangle$, is quite small for low F_ρ — less than 0.05 pixels (a reasonable average measurement uncertainty in the context of the displacement dynamic range of 0 to 5 pixels). It increases somewhat steadily from the minimum of 0.025 pixels at $F_\rho = 0\%$, until it seemingly changes slope precipitously for high F_ρ. This functional behavior is duplicated for the fluctuation standard deviation, $\langle \sigma_{LSD} \rangle_q$. This again suggests a possible F_ρ cutoff at about 25%.

Additional information about the influence of the filtering process upon the statistical uncertainty is displayed in Figure 29. All displayed information actually corresponds to $F_\rho = 0\%$, with the artificial horizontal offsets to allow for differentiation between the two velocity components. That is, the plotted symbols should actually be displayed directly as overlying the $F_\rho = 0\%$ line. It should be realized that there is little difference between the results for the two velocity components. The intriguing observation here is that $\langle q_{LSD} \rangle$ and $\langle \sigma_{LSD} \rangle_q$ have nearly identical low values after each of the filtering processes (the centered symbols literally overlap and obscure each other nearly exactly). As before, this indicates that this final velocity field has an average measurement uncertainty of about 0.025 pixels, with an approximately equal standard deviation.

For comparison, Willert & Gharib (1991) display "RMS fluctuations" for an equivalent ρ_P ranging from about 0.02 to 0.10 pixels (corresponding to an approximate mechanical displacement range of 0.01 to 10 pixels). Hence, our figure is at the least

comparable to theirs. Nonetheless, ours is an average over the entire displacement dynamic range *and* over the whole velocity field, rather than a specific low value that only occurs at a correspondingly low displacement.

Accuracy vs. uncertainty

A final statement can be made regarding the relationship between the figures of $\langle q_{LSD} \rangle = 0.025$ pixels and $\xi_{90\%} = 0.06$ pixels for the $\rho_P = 5.27$, $F_\rho = 0\%$ case. The key observation can be perceived from inspection of the iso-accuracy plots of Figure 27. It has been stated beforehand that the $\xi_{40\%}$ and $\xi_{80\%}$ values for this case were 0.02 and 0.05 pixels respectively. Further, it is possible to distinguish that $\xi_{50\%} \approx 0.02\text{-}0.03$ here. Thus, *half* of the vectors are within about 0.025 pixels of the theoretical result, a quantity numerically equal to $\langle q_{LSD} \rangle$ (which is *not* calculated from the theoretical velocity field, but is based on velocity *LSD's*). Additionally, the value of $\xi_{80\%}$ is within one standard deviation $\langle \sigma_{LSD} \rangle_q$ ($= 0.025$ pixels) of $\langle q_{LSD} \rangle$, with $\xi_{90\%}$ only slightly higher.

The foregoing confirms that a *majority* of the velocity vectors are measured to a significantly higher accuracy than indicated by $\xi_{90\%}$, with *nearly all* of the vectors within one standard deviation of the average measurement uncertainty of the velocity field. It would appear that $\xi_{90\%}$ is a useful measure of the capability of the PIV system precisely because it is actually a somewhat conservative indicator of accuracy.

3.3.2. Variation of particle intensity

W-component determination

In this simulation, the artificial particles of equations (38) and (39) had their peak intensities, I_0, randomized in both the first and second images. Barring other effects such as particle size or shape variations, the intensity change of a particle from one image to the next would indicate a transverse or w-component of the flow. Alternatively stated, a particle fading or brightening would indicate a flow into or out of the laser sheet.

If the laser sheet is assumed to have a thickness $z_0 = 1$ pixel, then its Gaussian radius would be 0.5 pixels, as shown by the following intensity distribution,

$$I(z) = I_0 \exp\left[- \frac{z^2}{(z_0/2)^2} \right] \tag{41}$$

An intensity derived w-component can then be defined by calculating z's corresponding to the random intensities I_1 and I_2 (obtained via inversion of equation (41)):

$$z_j = \pm \frac{z_0}{2} \sqrt{\ln \frac{I_0}{I_j}} \quad , \quad j = 1,2 \tag{42}$$

and dividing the difference between transverse locations by Δt,*

$$\left.\begin{aligned} dz &= z_2 - z_1 \\ d\hat{z} &= \frac{dz}{z_0} \end{aligned}\right\} \quad \hat{w} = \frac{d\hat{z}}{dt} \tag{43}$$

where the w-component has been normalized by z_0, which assigns it units of "laser sheet thicknesses per second." This removes dependence on the sheet thickness parameter, z_0,

---

*The sign ambiguity of equation (42) is resolved by using the *smallest possible* value for $(z_2 - z_1)$ as the basis for the w-component in equation (43). This is a worst case scenario, representing that even a smaller w-component can introduce degradation in velocity measurement.

yet permits an evaluation of an actual experimental setup given such a thickness. That is, the thickness of a laser sheet could be varied to correspond to an expected flow w-component.

These w-components were calculated for every particle in an image pair with $\rho_P = 5.27$. After processing the image pair using $(R,S) = (27,15)$ to obtain the 2-D velocity field, the *area-averaged* (over sample window) w-components were calculated from the average of the absolute values of the w-components of all particles within each sample window.* The absolute values were used to avoid possible near-zero average w-components resulting from particle w-components of opposite sign.

The resulting sample-averaged w-components were then normalized with a representative 2-D velocity: each sample window w-component was ratioed with that sample window's speed based on the *local* u-/v-components.† A final observation is that this "dimensionless" w-component ratio,

$$\frac{\hat{w}}{\sqrt{u^2 + v^2}}$$

actually has units of "laser sheet thicknesses per pixel."

Results

There were actually two image pairs used: one with constant peak intensities; and one with randomized peak intensities as discussed previously. The differences between

---

*As expressed before, $(R,S) = (27,15)$ results in a 33 × 33 velocity vector grid.

†This work has been duplicated using the flow field *average speed* (based on the overall average u- and v-components) for normalization. The equivalent figures for each one based on *local speed* will be indicated in footnotes.

the constant and randomized peak images resembles that of the previous studies. Namely, the degradation of velocity accuracy due to intensity variation is similar to the adverse effects produced by lowering ρ_p or increasing F_ρ. The values of $\xi_{90\%}$ were 0.17 and 0.34 pixels for the normal and randomized image *raw* velocity fields. These became to 0.06 and 0.11 pixels respectively after the combination filtering process, suggesting that filtering can compensate to some degree for this problem. The average w-component to speed ratio was approximately 0.075. Thus, if the laser sheet is 1 pixel thick, the average w-component ratio is 0.075; but if the thickness is 2 pixels, this average w becomes 0.15 or 15% of the local flow speed.*

The remainder of the analysis focused on the randomized images. The most conclusive data are displayed by Figure 30 and especially Figure 31, representing the raw and final velocity field results respectively.† Each symbol on these figures represents a plotting of the calculated dimensionless w-component versus the u-component subpixel accuracy *for one sample window.* There are a total of 1089 points on the graph, which corresponds to the chosen 33 × 33 velocity vector grid.

The subpixel accuracy was determined in the usual sense: through comparison with the theoretical u-component predicted for that sample window. For example, the leftmost point on Figure 31 indicates that the particular sample window experienced an average $\hat{w} \approx 0.075$ (or 7.5% of the local flow speed) and had a u-component measured

---

*This assumes that the laser sheet was not deliberately arranged to have a nongaussian intensity distribution.

†The equivalent figures using normalization of w-component by *average speed* are Figure 37 and Figure 38.

to be about 1.5×10^{-4} pixels from the theoretical value. Note that the displacement range on the previous $p(\xi)$ and $P(\xi)$ plots — from 0 to 0.5 pixels — approximates the interval bounded by 10^{-2} and 0.5 pixels on Figure 31.* Note further that most of the subpixel accuracy measurements lie within this interval. Also, the choice of u-component displacement difference field was arbitrary; the v-component data could as easily have been selected with negligible discrepancies.

The premise of this procedure was to establish a correlation between w-component and accuracy. It was found from the $P(\xi)$ data (not shown here) that a w-component introduced by randomization of particle intensities definitely degraded the overall accuracy. This suggested the possibility of a possible w-component threshold, similar to the previously suggested tolerance of $F_\rho = 20\text{-}25\%$ regarding out-of-plane particle motion. The working hypothesis at this point was that larger w-components would degrade accuracy more than smaller ones.

Neither Figure 30 nor Figure 31 support the idea that such a w-component threshold exists.† It can be clearly seen that a sample window having a particular subpixel accuracy can experience a variety of "dimensionless" w-components ranging up to 0.20 and higher. Hence, we found *no direct connection* between w-component and accuracy under these conditions. This does not contravene the reduction in accuracy of the entire flow field described earlier; it simply states that further analysis did not yield

---

*Naturally, 10^{-2} or 0.01 pixels is not the true lower bound of the $p(\xi)$ and $P(\xi)$ plots. In those plots, all measurements having a better accuracy than 0.01 pixels were nevertheless "labeled" as having that accuracy. This is because the "bin" size of the density function was chosen to be 0.01 pixels.

†Note that the filtering process (from Figure 30 to Figure 31) merely shifts the data to lower values (and better subpixel accuracy) without changing the nature of the data.

a more fundamental connection.

Additional information is presented in Figure 32 and Figure 33, with each converting the data of Figure 30 and Figure 31 respectively into histograms based on dimensionless w-component (recall that these are the raw and final velocity field data, respectively).* For example, the solid line corresponds to $\xi_{MAX} = \infty$, which signifies that the entire data set — ie., all the points on Figure 30 or Figure 31 — was converted into histogram form based on the ratio of w-component to local speed. Conversely, the $\xi_{MAX} = 0.20$ pixels curve indicates that only those points that had a subpixel accuracy better than 0.20 pixels were considered for the histograms. The notation "$\Sigma_\xi = .819$" on Figure 32 means that 81.9% of the data had such subpixel accuracy (this can be alternatively stated as $\xi_{81.9\%} = 0.20$ pixels). These fractional sums arise from the normalization of all histograms by the *total* number of points in the data set.

Observe that the histograms of Figure 32 and Figure 33 are heavily weighted to the smaller ratios of w-component to local speed. For ease of inspection, these figures have therefore been replotted by cutting off the higher two-thirds of the scale and expanding the remainder. These rescaled versions are displayed in Figure 34 and Figure 35.

Note the marked improvement of the histograms between Figure 34 and Figure 35. For instance, inspection of the $\xi_{MAX} = 0.20$ pixels curve for the final velocity field on Figure 33 shows "$\Sigma_\xi = .927$" or $\xi_{92.7\%} = 0.20$ pixels. Thus, 92.7% of the final velocity field was accurate to within 0.20 pixels, whereas only 81.9% of the raw velocity

---

*The equivalent figures using normalization of w-component by *average speed* are Figure 39 and Figure 40.

field was at that level of accuracy.

The hypothesis of larger w-components resulting in more degradation of accuracy would require that the histogram peak to "shift" to lower values as the displacement error cutoff ξ_{MAX} was decreased. It is evident that this is not the case: the histograms clearly stay centered at about $\xi = .075$, which is the aforementioned *average* w-component ratio.

This is made even clearer by Figure 36, which plots the histogram peak (ordinate) and histogram average (abscissa) as a function of ξ_{MAX}.* The histogram peak is continuously increasing, as was evident in both Figure 32 and Figure 33. The histogram average is remaining virtually constant, again at a value between 0.07 and 0.08. It is also evident from consideration of Figure 33 that the histogram peak abscissa, were it plotted, would also be close to constant. This confirms the conclusion that there is no further correlation between "dimensionless" w-component and subpixel accuracy beyond the obvious degradation over the entire flow field.

Final note on simulations

The results of the foregoing simulations defined a useful operating regime of our PIV implementation. The various thresholds were employed to initially determine the suitability of a real flow for examination, then to size the smallest possible sample windows for interrogation of that flow. These issues are explored further in §4.

---

*The equivalent figure using normalization of w-component by *average speed* is Figure 41.

4. REAL FLOWS

Background

As expressed earlier, we do not seed our experiments with extraneous particles. The expectation was that the resulting digitized flow images would require some sort of enhancement. This is because dirt or bubbles are not so easily distinguishable from the flow background. In fact, we have done some testing with a variety of preprocessing techniques: the approach has been to examine the intensity distribution of a flow image to determine a cutoff intensity; then set all intensities below the cutoff to zero. This process may or may not involve saturating the cutoff intensity.

This image preprocessing work has not yet proceeded to the establishment of a reliable procedure. The following sections will therefore discuss data from *raw* images only. That is, the PIV technique was still sufficient to interrogate even these lesser quality flow images. It is essential to realize that PIV is designed to follow specific *patterns* from one image to the next; if these patterns — usually consisting of a group of particles — do not change significantly between images, there is a better chance of a high correlation (and therefore, a reliable velocity vector measurement). In many cases, even the unenhanced images contained many regions where such "patterns" were recognizable. This process of pattern recognition can only improve with implementation of image enhancement techniques.

Note on image composition

The digitized images actually contain information corresponding to two different times. Each image *frame* is captured at the video rate, 30 Hz, but consists of odd and even *fields* which are separated by 1/60 seconds. Each field has 256 rows, with alternating rows corresponding to different fields.

One of these image frames therefore may contain two different locations for the same particle, reminiscent of a photographic image generated by a double-pulsed laser PIV system. The difference is that, for a particular particle, each field of the image has different parts of the same particle. Further, these two "half-particle" images may have noticeably different shapes. The counterpoint is that two such consecutive image frames containing both fields are more readily processed than two images having only odd or even fields (ie., one field is taken out and the missing rows are replaced by linear interpolation). The presence of odd and even fields within a frame artificially doubles the particle density (ρ_p) because the same particle is recorded twice. As expressed previously, higher ρ_p indirectly increases the available resolution of the PIV technique.

The limitation inherent in the use of dual-field image frames is that PIV — already a spatially and temporally averaging (over 1/30 seconds) technique — now becomes further temporally averaged over 1/60 seconds for the digitization rate of 30 Hz.* This is not crucial for slow flows like the following wake-shear layer flow, but it would be a severe constraint for rapidly changing flows. Images consisting of interpolated odd or even fields are also used in the following section, at a definite cost

*Of course, this matter becomes moot if the imaging is accomplished with a non-interlaced camera.

in resolution. Short of improving the framing rate, a method for at least mitigating the loss of resolution is the use of an overlapping sample window grid (this capability is currently being established). Examples of velocity fields using both kinds of source images will be discussed in §4.1.

4.1. Wake-shear layer flow

4.1.1. Flow/PIV parameters

A wake-shear layer flow was generated in a 4 cm wide test section with the two free streams (of water) moving at nearly equal flow rates. Under these conditions, the flow was *closer* to that of a wake than of a shear layer. These flow rates were in fact below the rated minimums of the flow meters, but extrapolation from the corresponding calibration curves yielded approximate velocities of

$$\begin{aligned} u_1 &\approx 6.4 \ cm/s \\ u_2 &\approx 5.4 \ cm/s \end{aligned} \tag{44}$$

The downstream distance from the splitter plate to the upstream edge of the image was 11.3 cm, with the flow going from right to left. The field of view was estimated by observing the number of pixels (out of 512) corresponding to the vertical height of 4 cm, with the assumption the pixels have nearly 1:1 aspect ratios. The resulting field of view was calculated to be 4.36 cm $\times$ 4.36 cm.

Figure 42 shows a sketch of the experimental setup, with the flow moving from right to left and the higher speed stream on the upper part of the test section. The actual flow image frames were sensed by an electronically shuttered CCD camera with an exposure time of 2 msec/field at the aforementioned 30 Hz framing rate. The image

frames were digitized and stored on a hard disk through a digital image acquisition/processing system (Trapix 5500). Two consecutive flow image frames were examined in order to determine an average pixel displacement between the two. Typical groups of particles were interactively selected and "followed" from one image frame to the next, with this difference in pixel locations being recorded. In this flow, the region having the faster stream was scrutinized because it would naturally have higher displacements. For this case, the registered displacement was about 21 pixels from frame to frame.

It was previously mentioned that Willert & Gharib (1991) attempted to resolve displacements of 10 pixels or less, due to restrictions upon their sample window size imposed by the Nyquist sampling criterion (cf. equation (3)). Placed in this context, a displacement of 21 pixels appears to be quite large. This is a direct consequence of our lack of sufficient temporal resolution: a smaller Δt between frames would have lowered the pixel displacement. We decided to process these image frames despite this constraint, and were nevertheless able to resolve the mean features of the flow.

We set $\Delta_{MAX} = 22$ pixels in order to encompass the displacement of 21 pixels in our experiment. The sample window size, S, was set semi-arbitrarily: the flow image frames were inspected to observe the smallest possible region that nevertheless contained at least 4 or 5 "particles," which complies with the criterion established by §3.2, as well as the recommendation of Lourenco & Krothapalli (1987). This was observed to be $S = 25$ pixels, which leads to a roam window size $R = 69$ pixels from equation (40). This corresponds to an 18×18 sample window grid, or 324 vectors. Together with the established field of view, $S = 25$ pixels yields a somewhat coarse resolution of

approximately 2 mm. It should be emphasized that this choice of sample window size was made for dual-field images; it became necessary to increase the sample window size to $S = 35$ for images consisting of an interpolated single field.

There are two issues to be identified from the previous choices. First, the Nyquist criterion limitation of Willert & Gharib (1991) is definitely not followed here; as expressed before, this limitation is not so severe in a non-FFT based PIV system. Nevertheless, the high value of Δ_{MAX} has forced a roam window that is substantially greater than its sample window. This greatly increases the computation time, since many more multiplications have to be performed (cf. equation (12)). The run-time of the software with these parameters is approximately *50 minutes* on a 25 MHz 80386-based personal computer with coprocessor. Recall that the run-time for the simulations of §3 with $(R,S) = (27,15)$ was only *7 minutes*. Improved between-image temporal resolution (with a correspondingly low Δ_{MAX}) would lower the run-time, along with the choice of the smallest possible (R,S) pair given the observed "particle" density.*

4.1.2. Velocity fields

PIV results — dual field images

Figure 43 shows the raw velocity field obtained from a consecutive image frame pair using the above parameters. The top two and bottom rows of the grid have been cut off: they corresponded to the boundary layer regions of the flow, and therefore could not be readily resolved because of the high velocity gradients. Note that there remain

*Also recall that a smaller S yields a higher resolution.

erroneous vectors within the remaining top and bottom rows. These appeared to have the same cause — being located within a region of higher shear — albeit to a lesser degree. There are notable edge effects at the right and left edges as well. These are caused by the Gaussian falloff of laser sheet intensity towards the sides. This is one of the details that should be addressable by image pre-processing.

Most of these effects were eliminated in the *combination* filtering process. It was necessary to interactively process a few additional points which resisted the initial replacement by the *LSD* filter because their neighboring points were also erroneous (for example, some of these were located in the leftmost two columns between 0 and 1 cm depth in Figure 43). The resulting filtered (final) velocity field is shown in Figure 44.

A better indicator of actual measured speeds is the constant velocity contour plot of Figure 45 (actually lines of constant u-component, since measured v-components were very small). The contour labels have units of centimeters per second. The shear layer nature of the flow is recognizable by large regions of nearly uniform velocity: 6.2 cm/s and 5.6 cm/s in the high and low speed regions respectively, with a definite velocity decrease in the intervening "wake" region. These values are close to the previously reported flow-rate derived estimates of 6.4 cm/s and 5.4 cm/s. However, one might expect that these regions should have even flatter velocity profiles. Possible causes for this velocity gradation will be suggested later. Note further that the effects of the erroneous leftmost edge points of Figure 43 could not be completely eliminated in Figure 45 (ie., the velocity contours corresponding to the two free streams should ideally traverse the entire span of the velocity field).

Velocity profiles

Profiles of the u-component of the velocity were obtained from Figure 44 and plotted for 3 downstream locations in Figure 46. This was done to suggest that the PIV technique had produced a plausible velocity profile. Observe that the regions of "flat" velocity are close to the estimated free stream speeds (cf. equation (44)). The measured velocity profiles do not appear to change significantly: the profile shapes seem nearly the same over a change in distance of roughly 2 cm downstream. Also, the lack of flatness of the free stream regions of the profiles can now be seen as having two conceivable causes: 1) the relatively low Reynolds number of the flow; 2) the possible lack of spatial resolution. The first condition suggests that the downstream location is such that the boundary layers from the test section walls and the wake itself have grown sufficiently to interact with each other. Alternatively stated, this implies that the inviscid core between the wake-shear layer and the boundary layer has ceased to exist.

The condition of inadequate spatial resolution could signify that spatial variations are simply being averaged over during the processing. Recall that decreasing S was not a viable option in this particular case because of insufficient observed "particle" density. In fact, lack of spatial resolution may not be the cause of this gradation in the velocity profiles. As will be shown next, a loss in resolution from $S = 25$ pixels to $S = 35$ pixels does not appear to degrade the profile very significantly.

PIV results — interpolated images

The previous situation worsens when using interpolated odd or even field images to eliminate frame-specific temporal averaging. Eliminating one of the fields is akin to

halving the effective particle density, necessitating a higher value of S to contain at least 4-5 particles per window. This choice of sample size can either be performed interactively as before, or by rerunning the software repeatedly for increasingly larger S until the number of erroneous velocity measurements appears sufficiently small. Beginning at $S = 25$, the latter procedure yielded a usable minimum $S = 35$, which yielded a 13 $\times$ 13 grid consisting of only 169 vectors and corresponds to about 3 mm resolution.

The final velocity field is displayed in Figure 47, with the top two and bottom rows again cut off to eliminate boundary layer regions from consideration. Figure 48 displays a comparison of the two PIV derived velocity fields as enacted by extracting velocity profiles from the same downstream location — 12.3 cm from the splitter plate. The decrease in resolution appears to have a minor effect upon the shape of the velocity profile.

Again, a way to improve the spatial data rate is to use an overlapping sample window grid. While the size of the sample window (and therefore, the *actual* spatial resolution) stays the same, hidden information may be obtainable between the existing grid points. This perhaps would give more definition to the measured velocity profiles. Velocity-derived quantities such as circulation and vorticity could then be calculated based upon the resulting velocity field. That information is already obtainable from the existing final velocity fields, but its reliability is in question due once again to the limited spatial resolution (ie., the velocity averages would be calculated over a too-large region). It seems apparent that a lowering of the time interval between image frames is needed for a actual enhancement of spatial resolution.

Part III. CONCLUSIONS

CONCLUSIONS

This section will summarize the makeup of our PIV system and the results obtained therefrom. These will be interspersed with recommendations and suggestions on increasing the versatility of the PIV technique.

The actual determination of velocity in our system is based upon the method of 2-D spatial cross-correlation, followed by 2-D Gaussian fitting around the displacement correlation peak to determine the displacement to subpixel accuracy. The processing time depends upon the sizes of the sample and roam windows. Given a sufficient particle density ρ_P, these windows can be downsized, which in turn reduces the computation time by lowering the number of multiplications per window cross-correlation. An example of this is the computation time of *7 minutes* for window sizes $(R,S) = (27,15)$.

The capacity for both nonoverlapping and overlapping sample window grids adds to the technique's potential versatility. A nonoverlapping application of the technique is faster, and gives us the ability to establish the general characteristics of the flow field and an optimal choice for S given the observed ρ_P. Then an overlapping grid can be chosen using this S, with the degree of overlap becoming a semi-arbitrary parameter. This would yield more details of the flow, to an extent more suitable for calculation of velocity-derived quantities.

The error processing of our current implementation nearly obviates the need for interactive identification of bad vectors, due to the *LSD*-based filtering process. The

choice of the N_{SDF} parameter determines the selectivity of the *LSD* filter. The next section will suggest a second parameter that may further reduce the necessity for interactive vector selection.

PIV in general enables ready calculation of velocity-derived quantities. Our own implementation has the capability of generating circulation and area-averaged vorticity, with the near-term addition of vorticity based on differentiation of velocity.

The accuracy of our implementation has been demonstrated on simulations having a wide velocity dynamic range, without needing excessive particle density. For the default $\rho_P = 5.27$, two measures of accuracy were constructed: 1) $\xi_{90\%} = 0.06$ pixels; 2) $\langle q_{LSD} \rangle = 0.025$ pixels. The first result indicates that 90% of the velocity measurements were accurate to within 0.06 pixels. The second one indicates that the average over the *entire* velocity field of the uncertainty of velocity measurement is 0.025 pixels.* This uncertainty value signifies that most of the velocity vectors are actually measured to somewhat better than 0.06 pixels, which suggests in turn that $\xi_{90\%}$ is a fairly conservative indicator.

Insufficient particle density is apparently the dominant source of erroneous velocity measurements, with strong velocity gradients and out-of-plane motion next in importance (cf. §2.1).† This conclusion arises because most other possible sources of error appear to indirectly degrade the *effective* particle density. It therefore seems that

---

*Alternatively expressed, this represents the average *local standard deviation* (*LSD*) of a velocity vector with respect to an *average velocity* (based on its eight surrounding vectors).

†The adverse effects of strong gradients are termed *velocity bias* by Adrian (1991) and are also mentioned by Willert and Gharib (1991).

ρ_P is the major parameter.*

In order to mitigate the effects of such errors, emphasis will be placed on the adaptability and versatility of the technique to operate within a set of guidelines. Sizing sample windows on the basis of $\rho_P \geq 4$ is the initial criterion. Limitation on out-of-plane motion seems the logical next choice, which in our case is represented by $F_\rho < 25\%$ (which is numerically close to the w-component threshold suggested by Adrian (1991)). Indeed, it is possible that simple inspection of an image data set can identify excessive out-of-plane motion, which can lead to quick modification of the experimental apparatus to minimize the effects of this motion.

The main challenge of our PIV system is to measure velocity despite lower quality images, which arise because we do not deliberately seed the flow with extraneous particles. Instead, we depend upon the naturally occurring dirt particulates in the flow.† The major set of guidelines will therefore arise out of the statistics of image pre-processing. Our "particles" are in fact small regions of high contrast in intensity (compared with the flow background). It remains to test techniques such as edge detection or binary thresholding for their efficacy in heightening this contrast. Some preliminary success has been achieved by simply cutting off all intensities below a given grey level, and scaling up the remaining intensities.

---

*Strong velocity gradients are simply not readily resolvable without the ability to reduce sample window size (thereby increasing resolution) such that the gradients can be neglected *within* the sample window. This in turn necessitates a higher ρ_P such that there are sufficient particles within even the reduced sample window. Keane & Adrian (1990) does contain some recommendations regarding maximum permissible velocity gradients within an interrogation window, based on their simulations.

†In the near term, our PIV system will also be tested on flows using hydrogen bubbles as particles. It is believed that image enhancement techniques developed for nonuniform dirt particulates would possibly be suitable for bubbles as well.

Such a fairly simple set of processing tools should suffice for initial experiments; constant modification will render these general routines to adapt to a greater variety of experiments. Nonetheless, it has been shown that even with a raw, unprocessed image pair interrogated using a nonoverlapping grid, the PIV technique was still able to reveal the main features of the flow, with apparently reasonable velocity measurements.

It should be stressed that this PIV implementation is by no means incapable of analyzing flows *with* seeded particles. On the contrary, such flows should be even easier to analyze. The particles of the simulations were designed specifically to mimic seeded particles in both size and contrast. A seeded flow would probably produce less velocity measurement error due to better definition of displacement correlation peaks. The crux of the matter is that seeding the flow is also an experimental complication which need not be necessary for our system to work.

FUTURE WORK

There are several possibilities worth investigating for improvement of all aspects of our PIV implementation. Many of these would be intended to increase the flexibility of the technique, while maintaining or possibly increasing the accuracy.

Image preprocessing

Digital image preprocessing is the most immediate modification, as has already been discussed. The development of a set of adjustable enhancement procedures is essential for interrogation of non-explicitly seeded flows. As expressed earlier, binary thresholding (with rescaling methods including varying degrees of saturation) has already

shown some promise; there are probably a plethora of additional techniques available within the image processing community. All that remains is their adaptation for use in a PIV context.

Error processing

A simpler modification would be in error processing, upon the *LSD* filtering process. Equation (20) represents the comparison of velocity differences, $|\delta q|$ with their velocity *LSD's*, q_{LSD} (where q can represent either u or v). It may be useful to perform these comparison in polar coordinates instead of the foregoing Cartesian coordinates, since it is conceivable that only one of the two component *LSD* comparisons would be necessary to identify erroneous vectors. This version would consist of comparing velocity magnitude *and/or* direction to their respective *LSD's*.

A major issue of interrogation of real flows is the possibility that there will be isolated regions of the flow containing several erroneous velocity vector measurements. The *LSD* filter may fail to detect these vectors since neighboring bad vectors artificially inflate each other's velocity *LSD's*. To be identified by the *LSD* criterion in this case, the velocity differences in equation (20) must then be excessively large in order to be greater than these inflated *LSD's*; this may not be the case for the regions under consideration.

A modification with the potential to detect such regions is a *secondary LSD* filter along the lines discussed in §2.3.2. It is designed to identify vectors with just such inflated *LSD's* via comparison to the average *LSD* of the entire velocity field (this average is the same as the previously mentioned average uncertainty, $\langle q_{LSD} \rangle$). The

following condition was executed to be analogous to equation (20), but using the definitions of equations (25) and (26):*

$$q_{LSD} > (N_{SSD} + 1) \times \langle \sigma_{LSD} \rangle_q \tag{45}$$

where: $N_{SSD} \equiv$ *secondary standard deviation factor*

N_{SSD} is a constant similar in application to N_{SDF} in equation (20); it is also arbitrarily selectable for a particular flow. Consider Figure 49, which represents the wake-shear layer flow discussed earlier. The figure axes are in (n_x,n_y) grid coordinates. The asterisk symbols represent vectors failing the primary *LSD* criterion ($N_{SDF} = 2$), and the plus signs represent vectors failing the secondary *LSD* criterion ($N_{SSD} = 1$). The indicated erroneous points may be visually confirmed through inspection of the visibly bad vectors of the raw velocity field as plotted in Figure 43.

It was found that ten points failed to be identified by the primary *LSD* criterion for the wake-shear layer flow of Figure 49 (ie., there should have been ten more asterisks in the figure). The grid coordinates of these points are listed in the table to the right, and can be checked against locations of bad vectors in the raw velocity field of Figure 43. Preliminary experimentation with the secondary *LSD* filter revealed that five of these vectors were able to be identified as bad vectors with this method (these are in **boldface**). Alternatively stated, five

n_x	n_y
11	1
15	2
16	6
1	8
0	**9**
0	**10**
11	10
17	**10**
0	**11**
1	**11**

*Equation (45) contains an assumption that all vectors with velocity *LSD's* within one standard deviation of the average velocity *LSD*, $\langle q_{LSD} \rangle$, are acceptable vectors. This assumption is readily adjustable, should the need arise.

of the plus signs in Figure 49 represent correct identification of erroneous vectors (that went undetected by the primary *LSD* filter). It is possible a value of $N_{SSD} < 1$ may well have succeeded in detecting all erroneous vectors unselected by the primary *LSD* filter. Regarding the wake-shear flow of §4, the remaining five points were nonetheless readily identifiable and were interactively replaced to produce the final velocity field of Figure 44.

As is true with image preprocessing, it would probably be prohibitive to attempt to design a universal procedure for error detection; the focus should be on making the error detection process as versatile as possible. This may well be accomplished with the *LSD* filter using the variability of the N_{SDF} and N_{SSD} parameters.

Velocity-derived quantities

All velocity-derived quantities are more reliable in a denser velocity vector grid, and so are better used with an overlapping vector grid. Differential vorticity is one such property currently being instituted as an option. It is possible to add several other methods for processing vorticity and especially circulation. For instance, the current method for circular integration of circulation is based on a quarter-circular contour. It is workable to diversify the circulation integration mechanism of Appendix C into a method for integration around an arbitrary contour. Recall from §2.4 that once the geometry is defined, the integration is readily achievable. Finally, this integration procedure could possibly be further improved through application of Simpson's rule or quadratures instead of the present 2-D trapezoidal rule.

Velocity field processing

In Willert and Gharib (1991), it is mentioned that the aspect ratio of the interrogation window could be changed from 1:1 to one more suitable for a particular flow. This also makes sense for our PIV application, although it would probably be better applied to the roam window instead of the sample window. Furthermore, some experimentation on bad vector detection *during* velocity processing could also be appropriate. The suggestion of Adrian (1991) of using a correlation peak detection threshold certainly has merit in this context (cf. §2.1). An additional possibility would be to check velocity measurements for marked deviation from the continuity equation.

The above error detection schemes need not be implemented independently of each other. The velocity field processing could be performed with simultaneous use of several schemes, with each scheme generating a list of erroneous velocity grid points. Those points that are detected by at least two error detection methods would be targeted for replacement. This would reduce the number of "good" points that are selected for replacement by a single error detection scheme.

Another useful modification would be the introduction of ***dynamic window sizing***. For example, a velocity measurement could fail a criterion — such as the correlation peak threshold — as a result of large flow displacement. The particles within the sample window could have traveled sufficiently far in Δt to have some of them disappear off the edge of the roam window. In such circumstances, the roam window size R could be increased by 2 pixels, and the roam and sample windows could be re-correlated. The roam size could be repeatedly incremented until a "good" measurement is obtained.

An extension of this concept would be to increment *both* window sizes instead of

only R. This would be more functional if the sample window has too few particles: the increment in size would presumably bring more particles into the window, and increasing S and R by the same amount would preserve the same maximum resolvable displacement.

Conceivably the most effective of all of the foregoing modifications would be the imposing of a ***2-D spatial perturbation*** upon the grid locations of suspect velocity vectors. This method would be most appropriate under the following conditions: 1) the sample window has too few particles; 2) there are sufficient additional particles *slightly beyond the bounds* of the sample window. In essence, the sample window grid was defined in a "dead spot" where there were insufficient particles in the flow; but a slight adjustment of the sample window location brings other particles in. This would presumably be detected by correlation peak criterion as mentioned above.

Instead of changing R or S here, the velocity correlation would simply be repeated at four points *closely adjacent* to the point being considered, as in Figure 50. If the center point had grid coordinates (n_x,n_y), then the four points would have coordinates based on $(n_x \pm \epsilon, n_y \pm \epsilon)$ where $\epsilon \ll S$ (ie., ϵ would only be about 1 or 2 pixels).

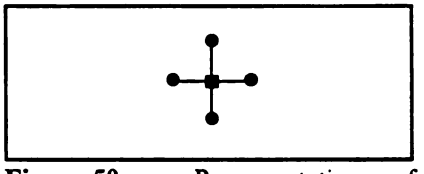


Figure 50. Representation of 2-D spatial perturbation around velocity vector grid point.

The vectors obtained from measurement of these four points would then be *averaged* and the result designated to the center point. This offers a kind of insurance against bad velocity measurements by making these measurements redundant over a small spatial area. Even a repeated erroneous measurement yields the confirmation that the particular flow region is indeed a significant trouble spot, for whatever reason.

Final comments

No specific flows have been heretofore mentioned in this section since the eventual goal of the PIV implementation is to investigate most flows. Work will nevertheless proceed on shear layers, with resolution of ever more complex vortical motions given preference. The dynamic stall regime of unsteady airfoil motion is another possibility for investigation.

The final conclusion is that this PIV system is definitely an early version; clearly there is much that can be done to improve the technique. The fact remains that it is already useful, even given unprocessed flow images to determine the general features of a flow (or to establish the inapplicability of PIV to the flow if it has insufficient ρ_p or excessive velocity gradients or 3-D motion).

Digital-based PIV as a field is certainly also in its early stages, and appears to be undergoing increasingly rapid growth. The technique should continue to experience advancement concomitant with advances in imaging technology.

FIGURES

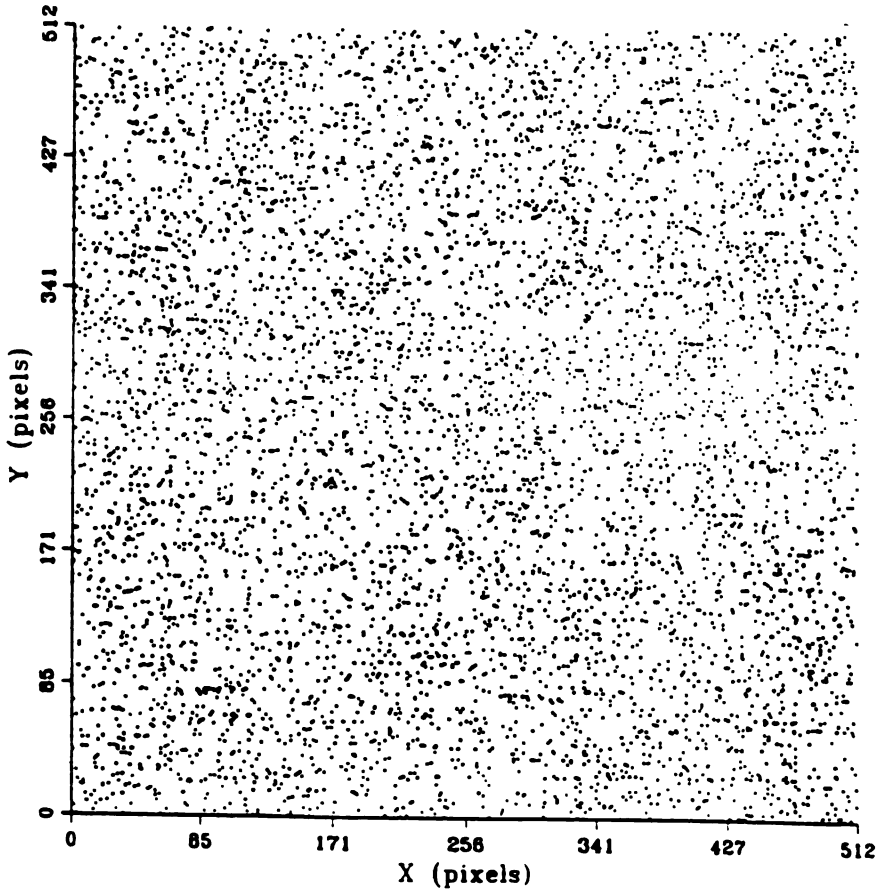
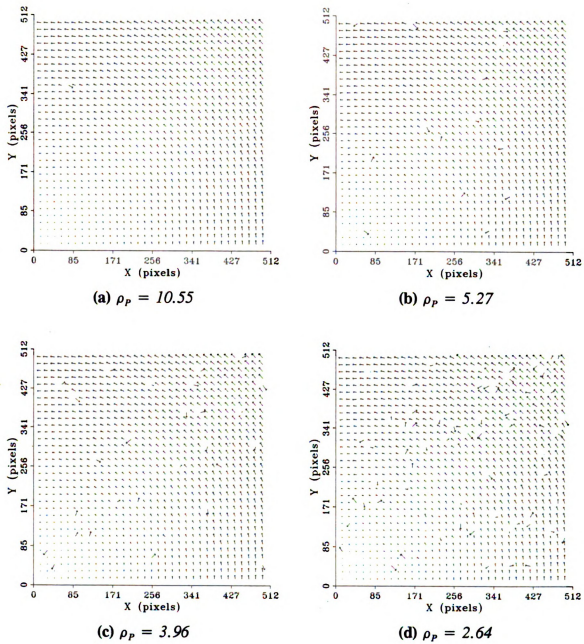


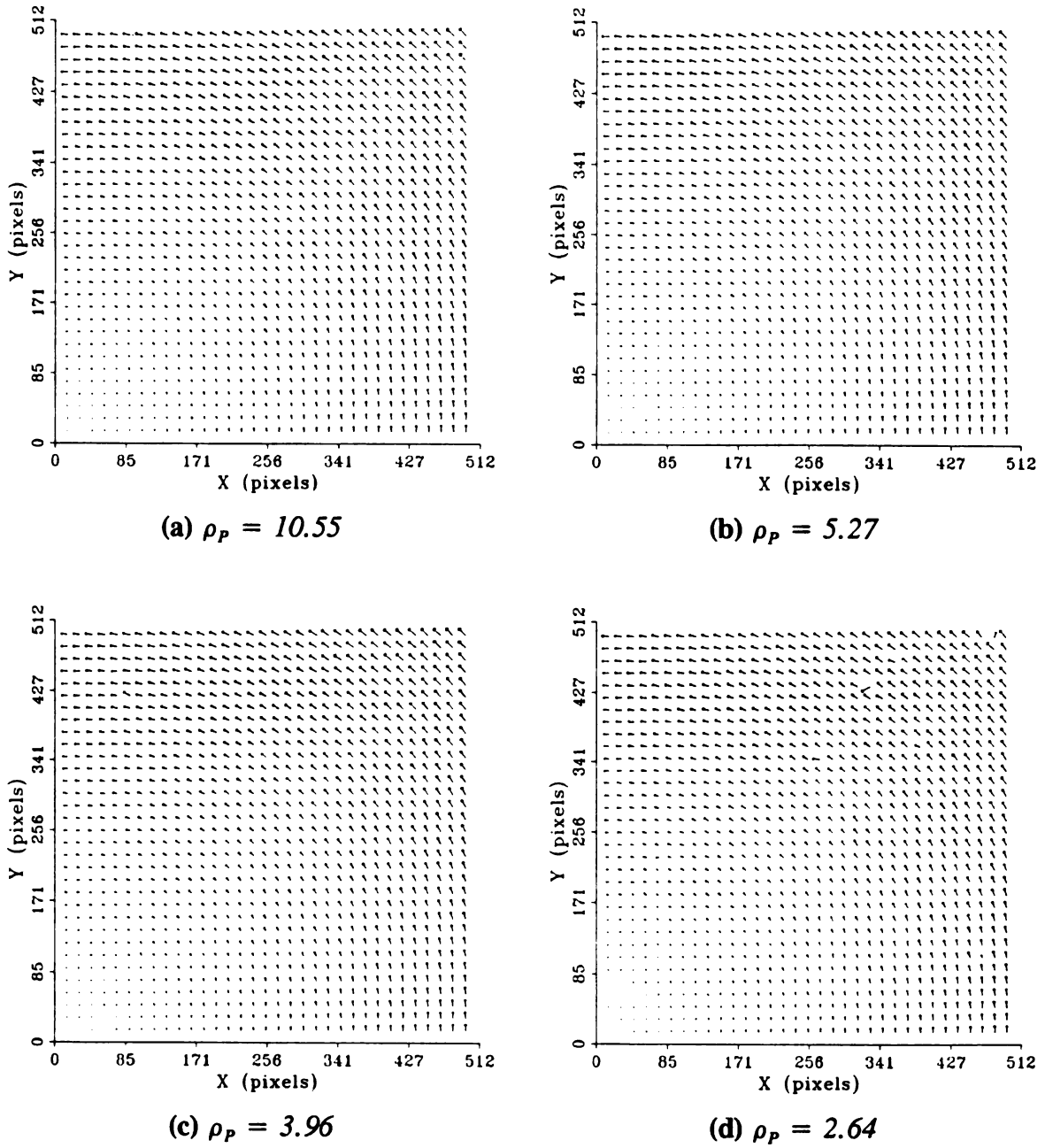
Figure 8. Seed particles randomly distributed over a 512 × 512 pixel^2 image plane.



(a) $\rho_P = 10.55$ (b) $\rho_P = 5.27$

(c) $\rho_P = 3.96$ (d) $\rho_P = 2.64$

Figure 9. Variation of ρ_P — (RAW) velocity fields.



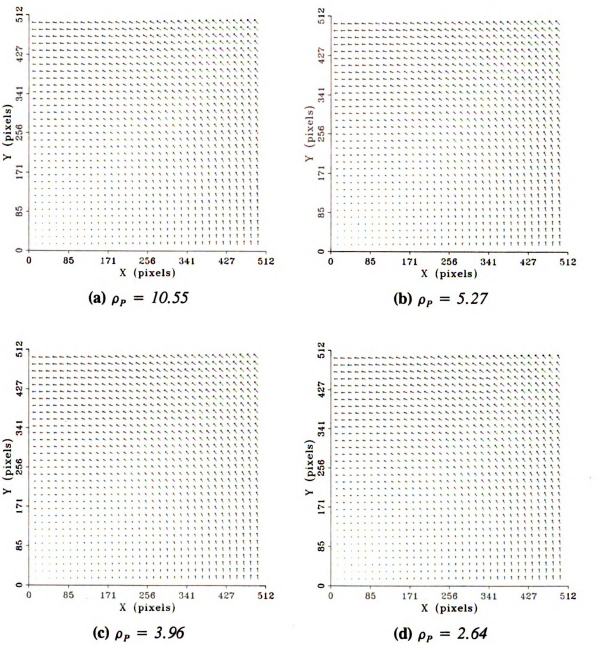
(a) $\rho_P = 10.55$

(b) $\rho_P = 5.27$

(c) $\rho_P = 3.96$

(d) $\rho_P = 2.64$

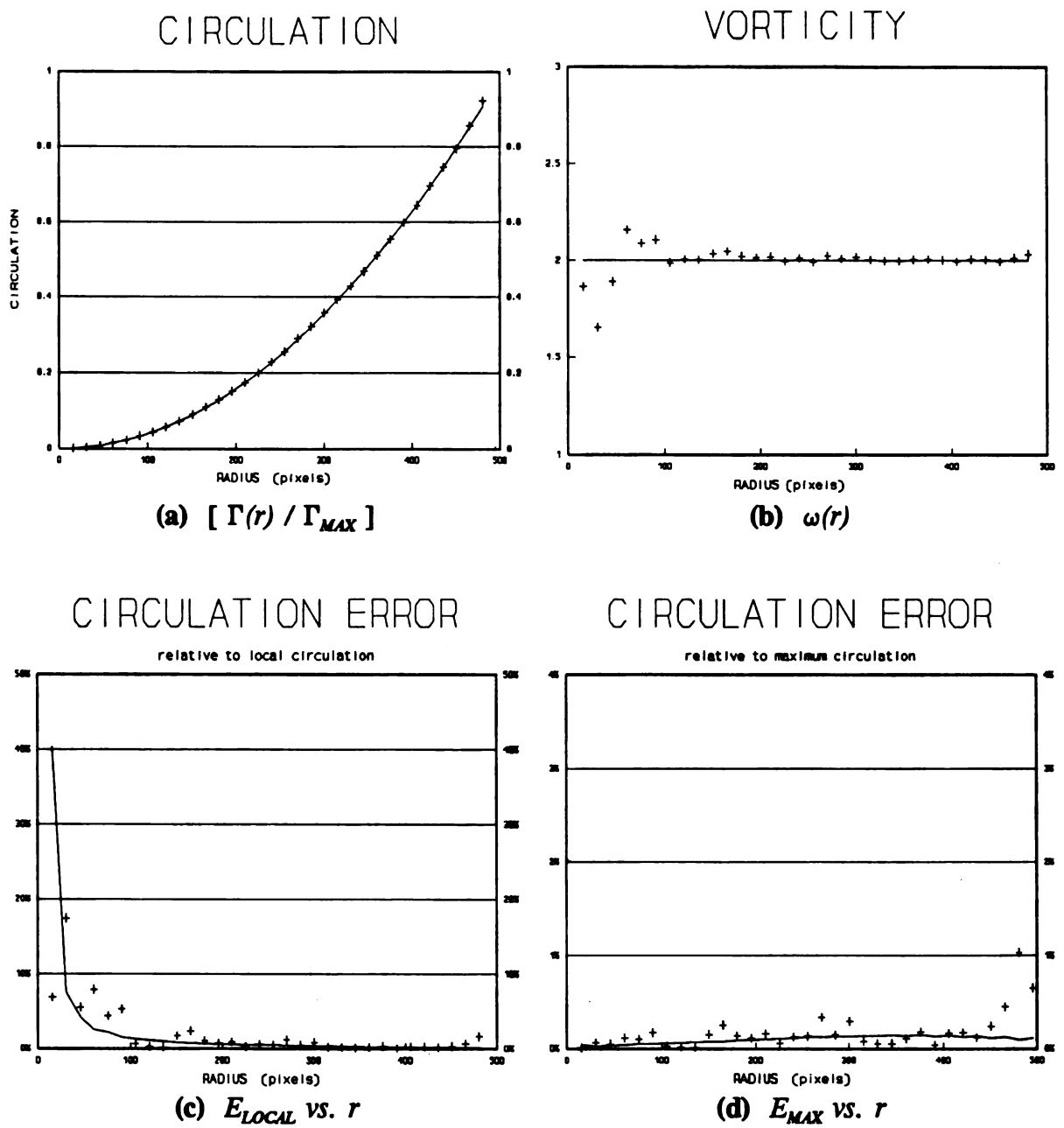
Figure 10. Variation of ρ_P — (LSD) velocity fields.



(a) $\rho_P = 10.55$ (b) $\rho_P = 5.27$

(c) $\rho_P = 3.96$ (d) $\rho_P = 2.64$

Figure 11. Variation of ρ_P — (FINAL) velocity fields.



(a) $[\ \Gamma(r) / \Gamma_{MAX}\ ]$ **(b)** $\omega(r)$

(c) E_{LOCAL} *vs.* r **(d)** E_{MAX} *vs.* r

Figure 12. Variation of ρ_p — velocity-derived quantities, $\rho_p = 5.27$ (FINAL).

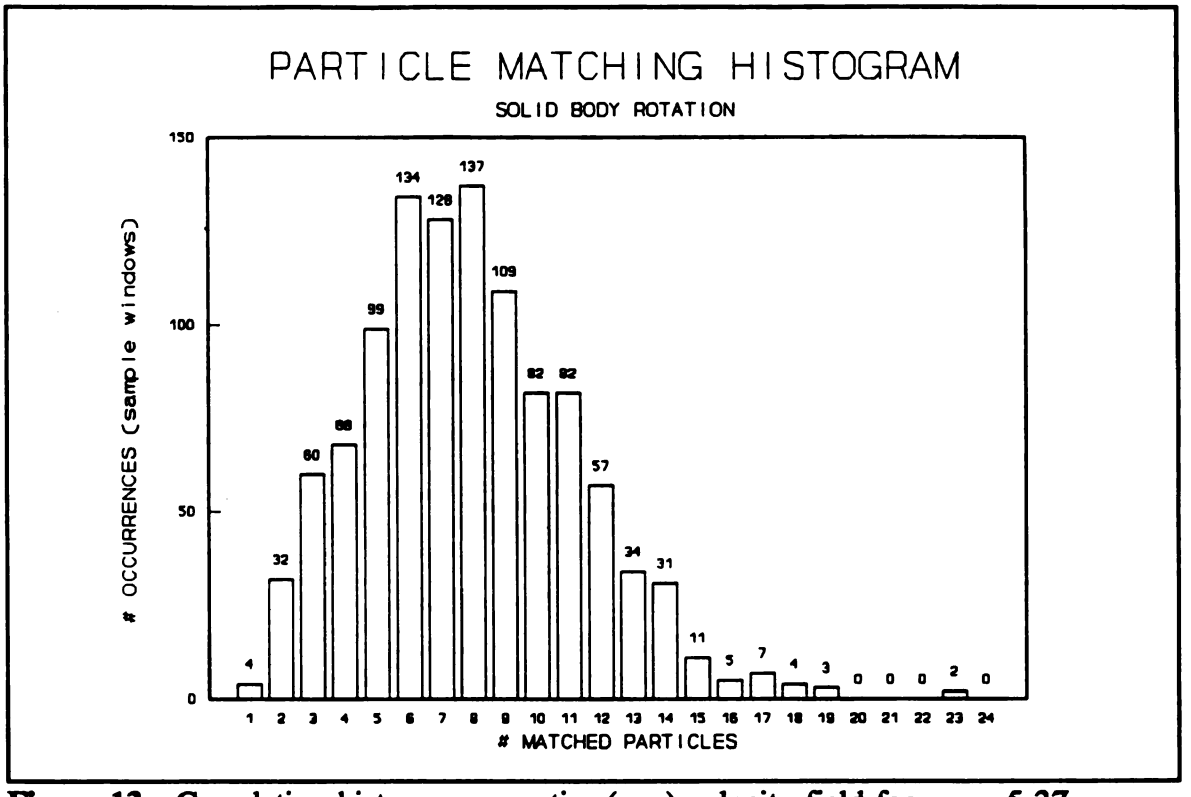


Figure 13. Correlation histogram — entire (raw) velocity field for $\rho_p = 5.27$.

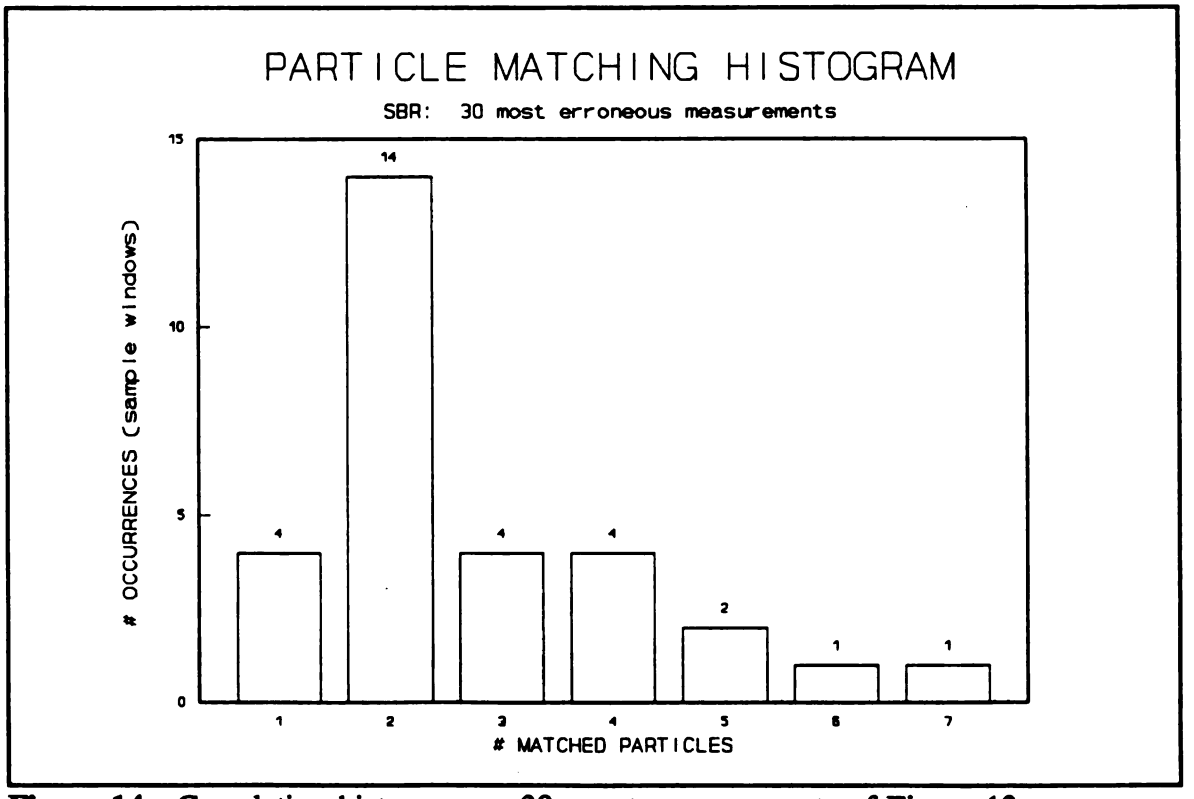


Figure 14. Correlation histogram — 30 worst measurements of Figure 13.

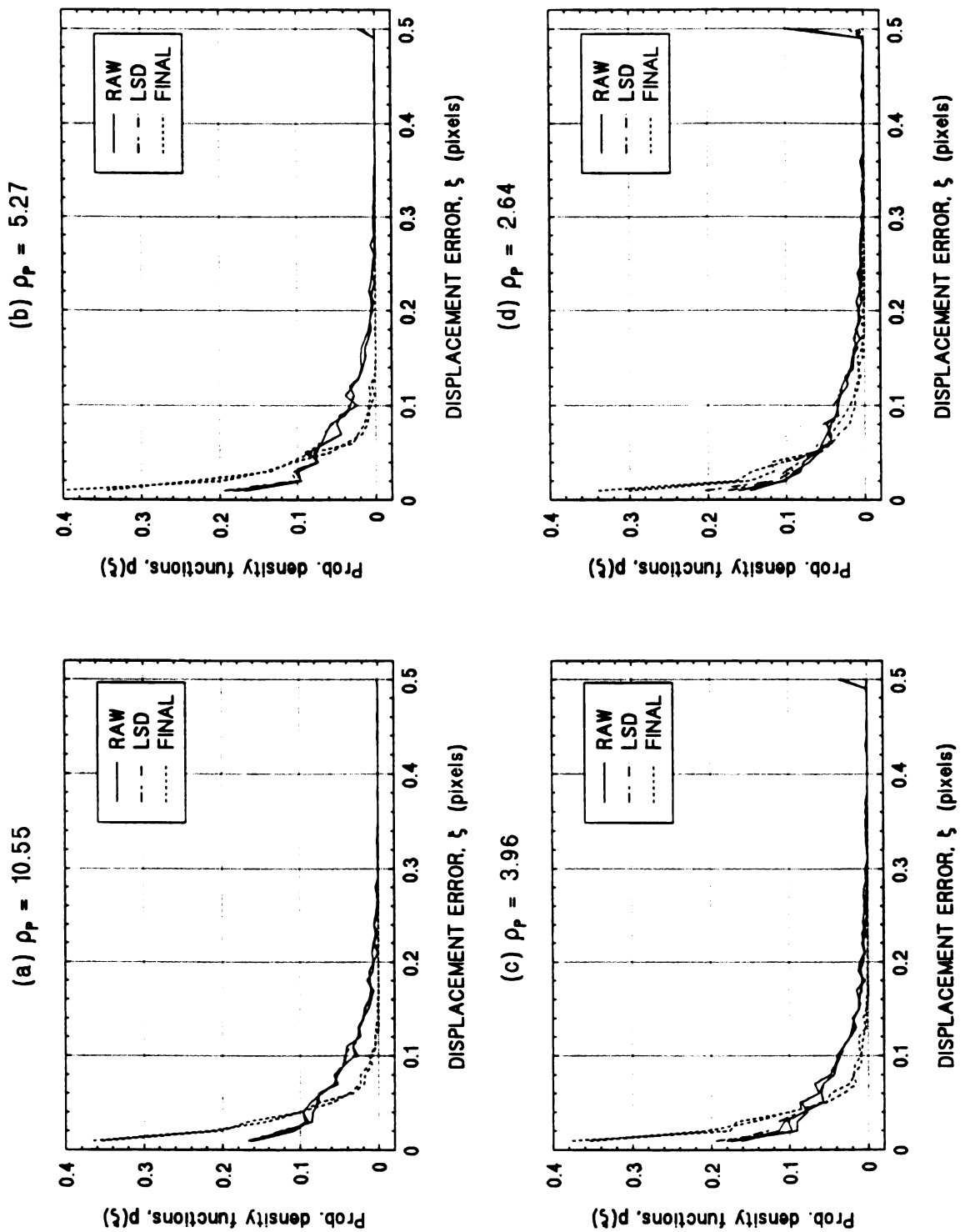


Figure 15. Variation of ρ_P — probability density functions, $p(\xi)$.

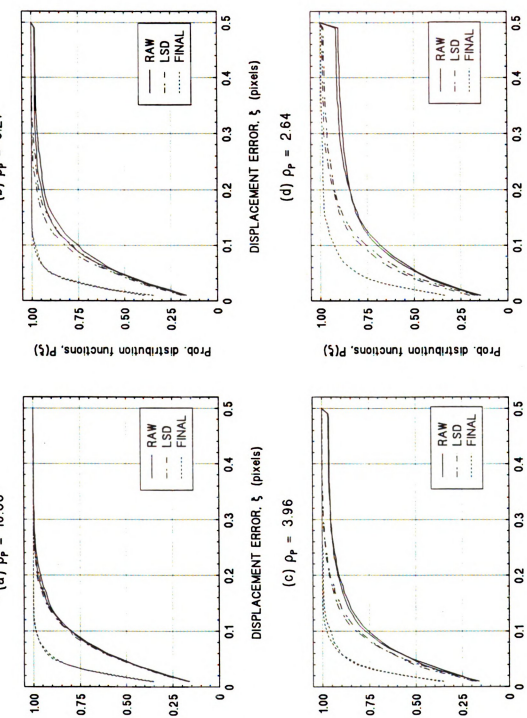


Figure 16. Variation of ρ_P — probability distribution functions, $P(\xi)$.

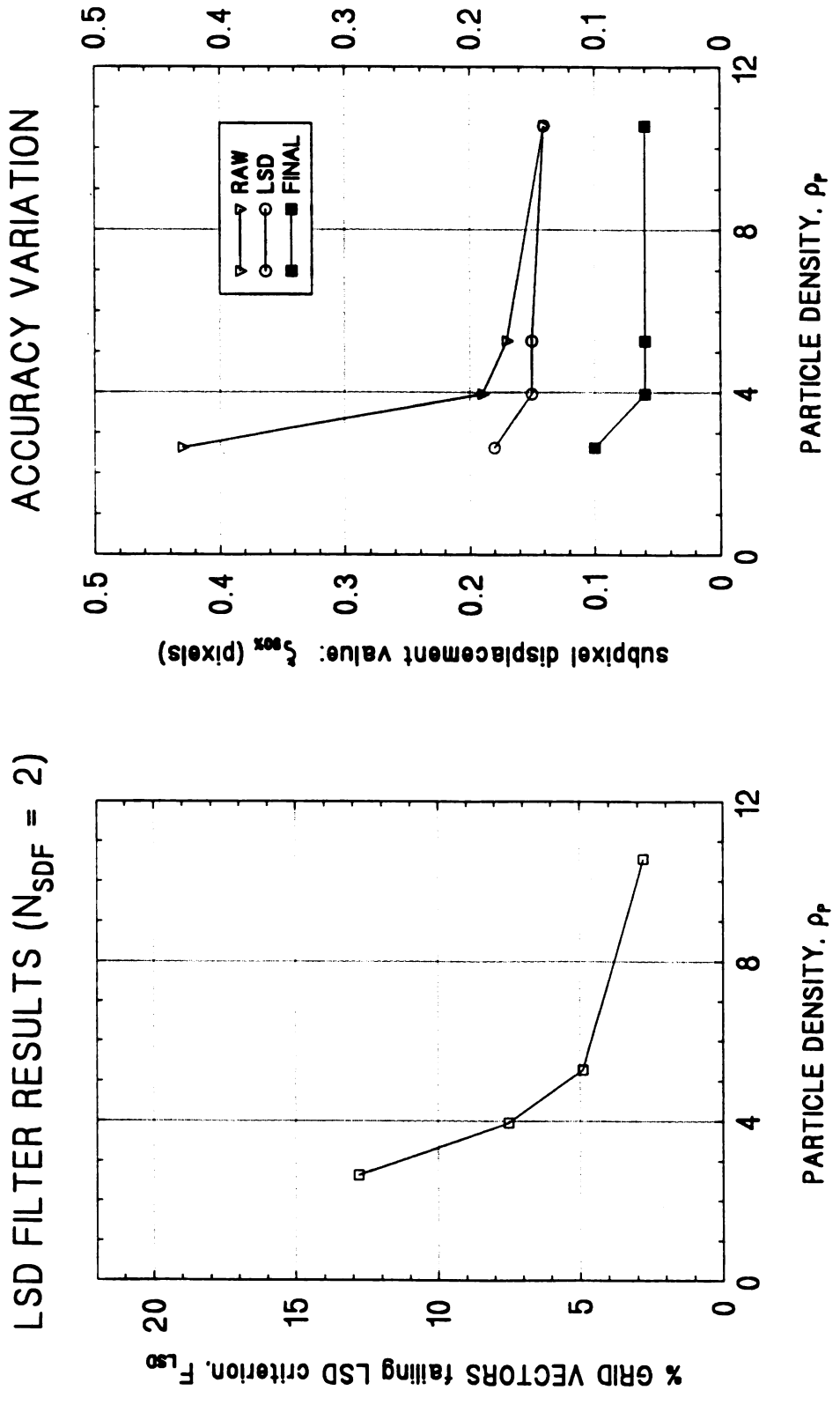


Figure 17. Variation of $\rho_P - \xi_{90\%}$ and F_{LSD} vs. ρ_P.

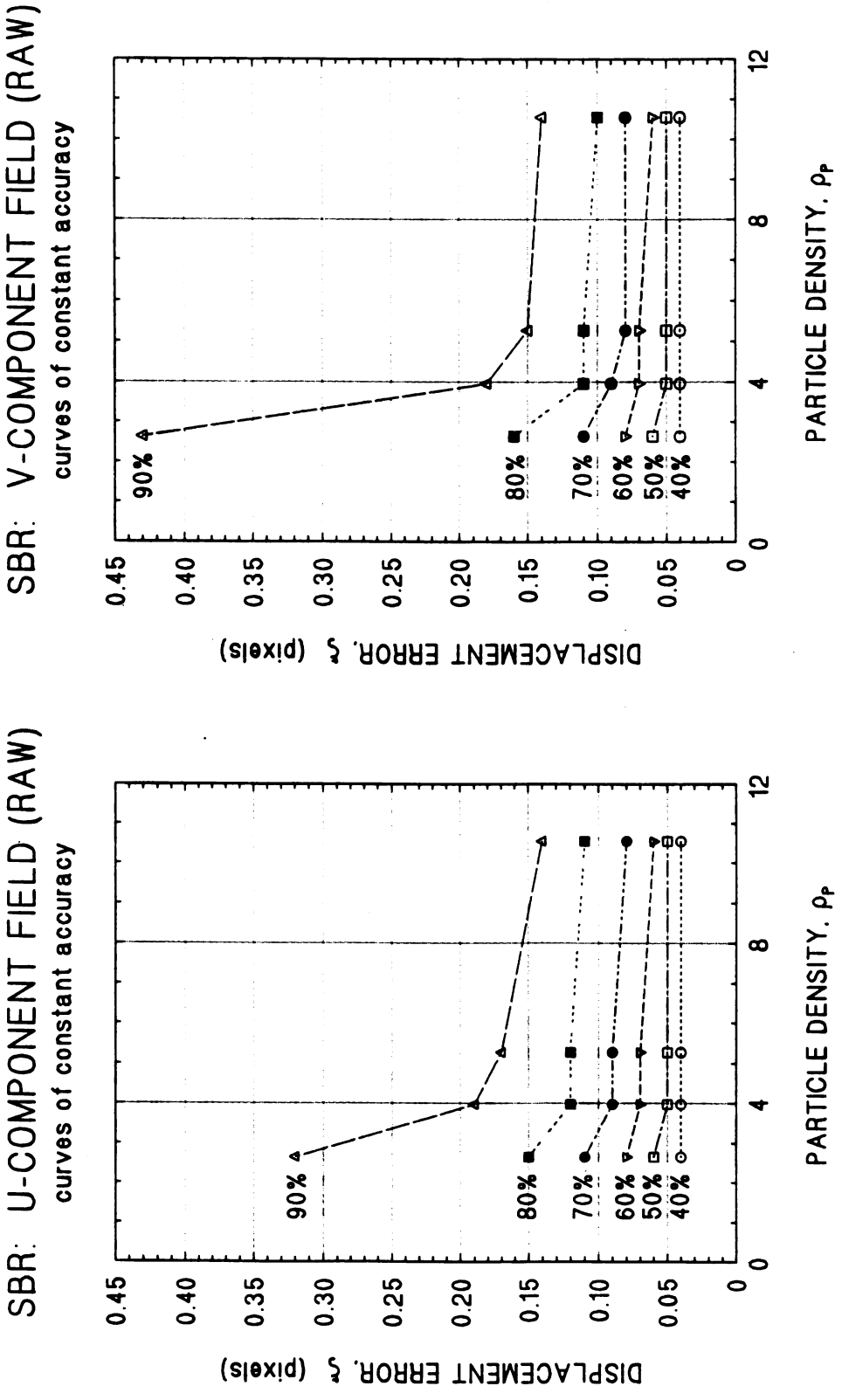


Figure 18. Variation of ρ_P — iso-accuracy plots (RAW).

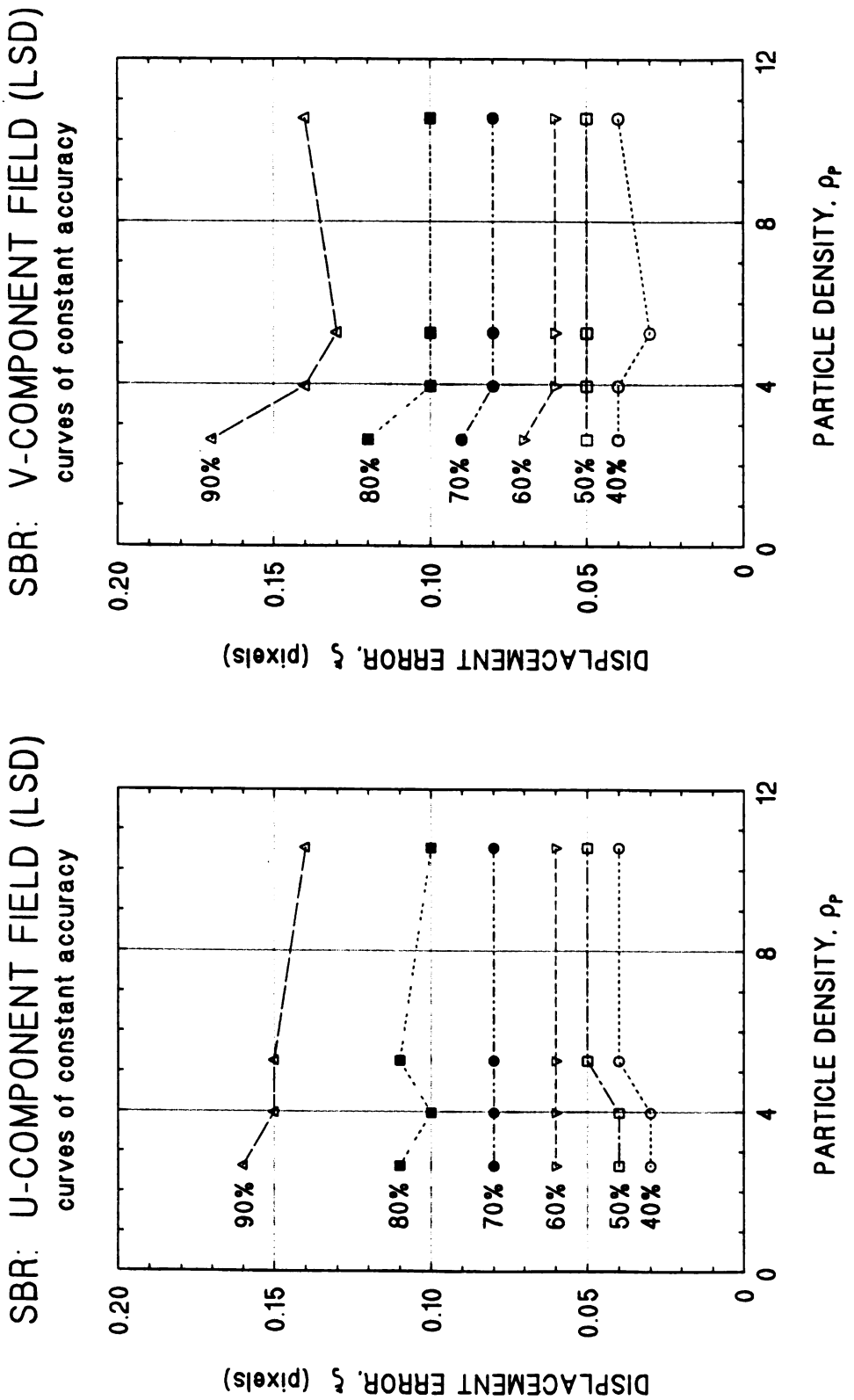


Figure 19. Variation of ρ_P — iso-accuracy plots (LSD).

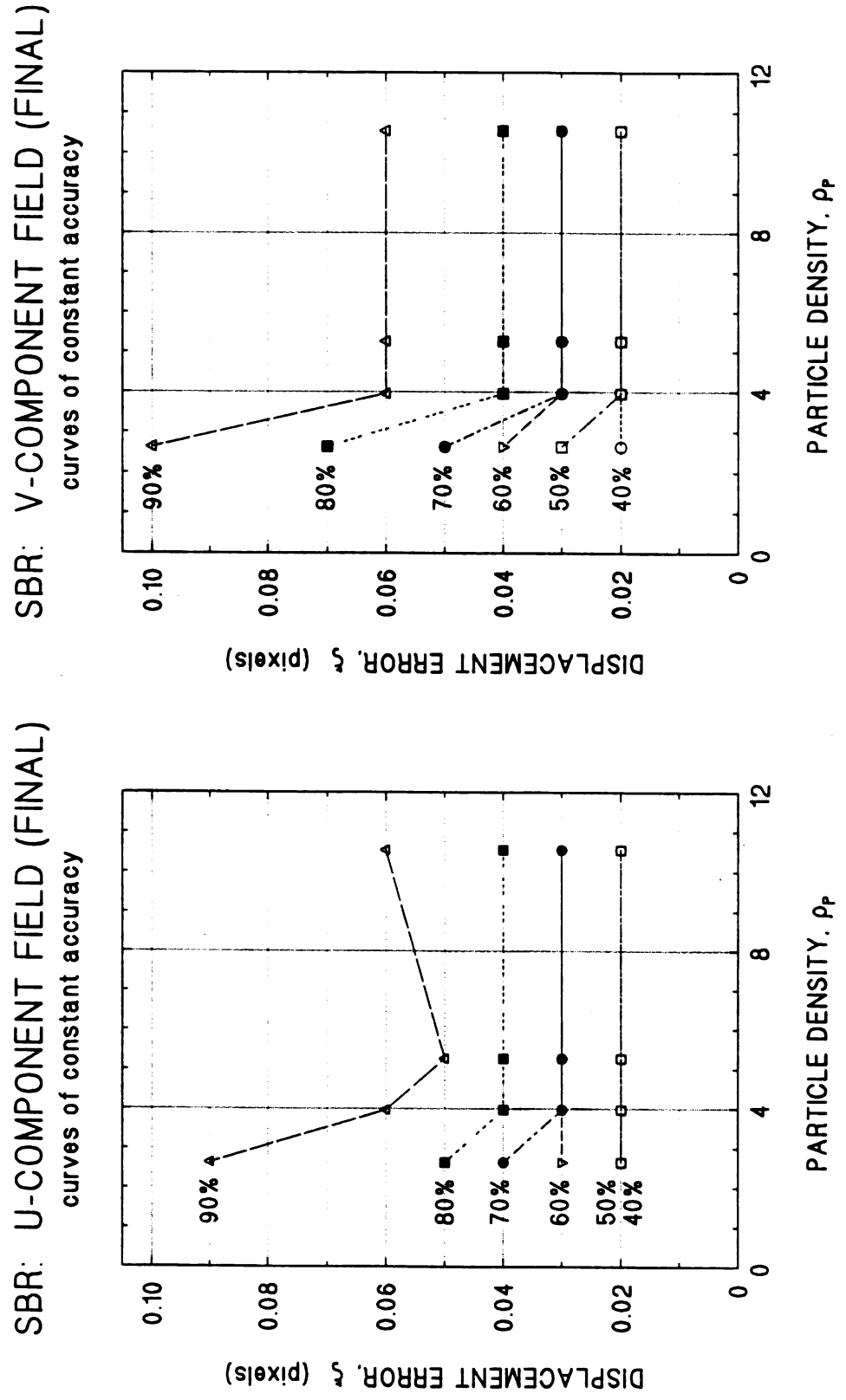
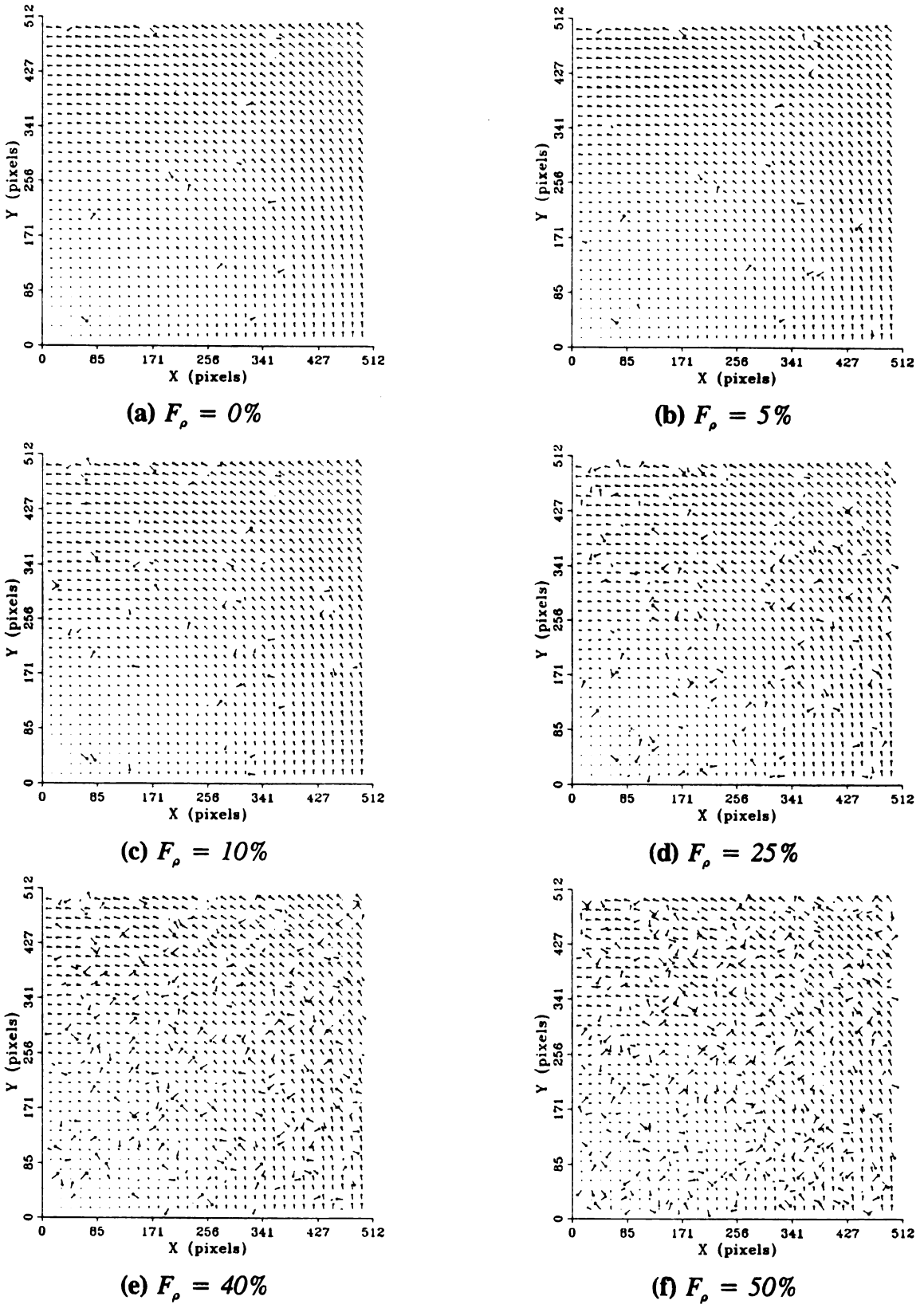


Figure 20. Variation of ρ_P — iso-accuracy plots (FINAL).

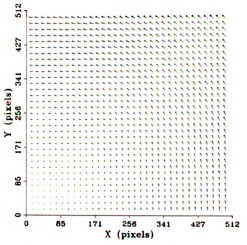


(a) $F_\rho = 0\%$ (b) $F_\rho = 5\%$

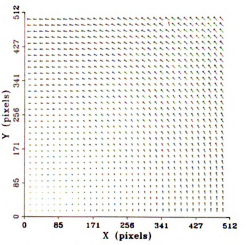
(c) $F_\rho = 10\%$ (d) $F_\rho = 25\%$

(e) $F_\rho = 40\%$ (f) $F_\rho = 50\%$

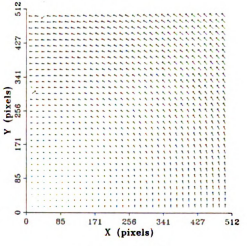
Figure 21. Variation of F_ρ — (RAW) velocity fields.



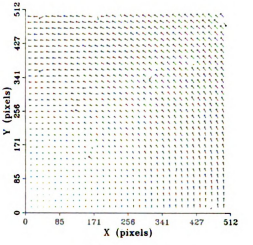
(a) $F_\rho = 0\%$



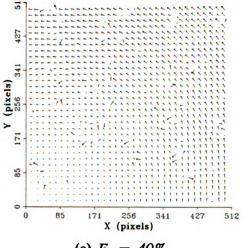
(b) $F_\rho = 5\%$



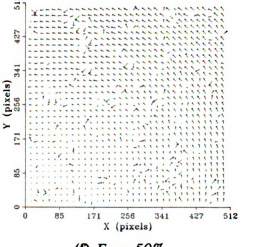
(c) $F_\rho = 10\%$



(d) $F_\rho = 25\%$

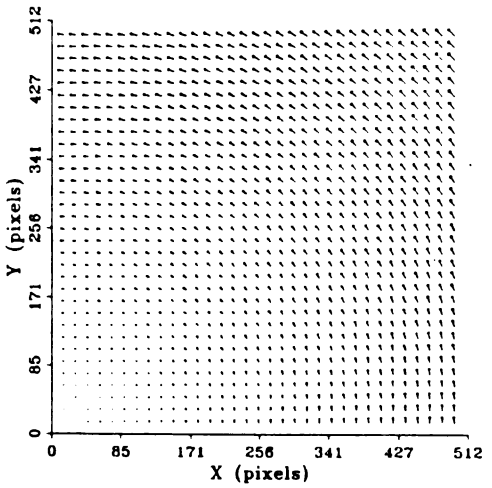


(e) $F_\rho = 40\%$

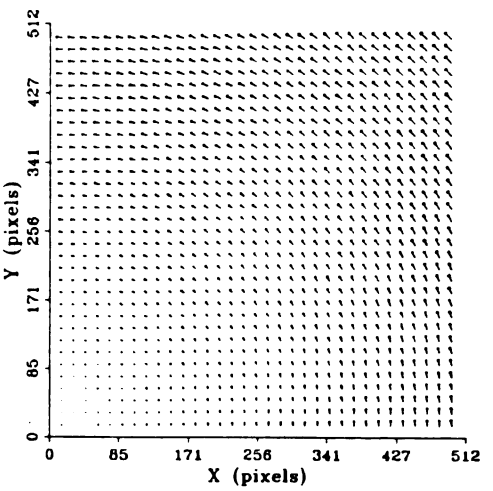


(f) $F_\rho = 50\%$

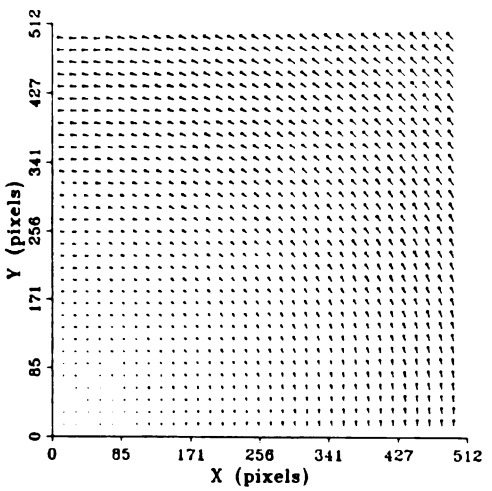
Figure 22. Variation of F_ρ — (LSD) velocity fields.



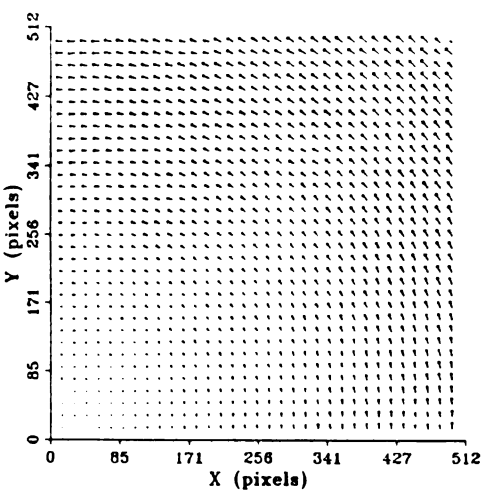
(a) $F_\rho = 0\%$



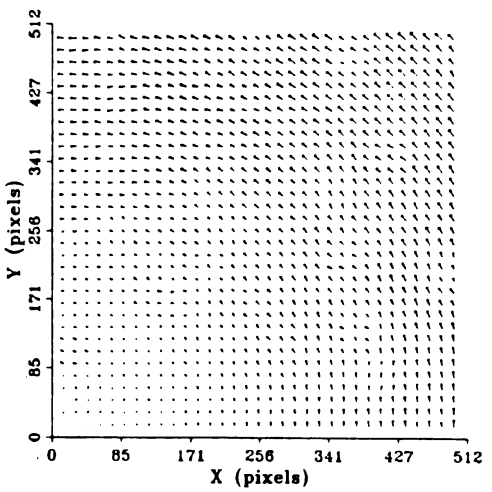
(b) $F_\rho = 5\%$



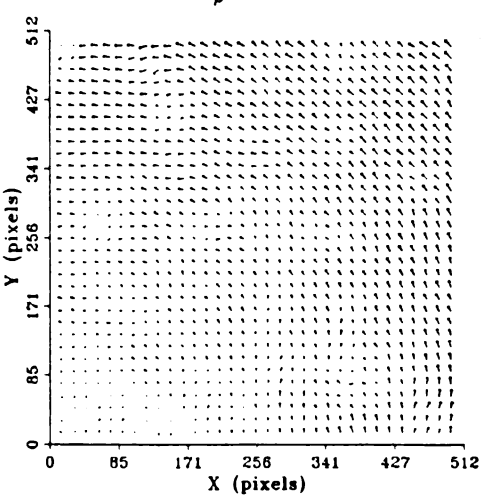
(c) $F_\rho = 10\%$



(d) $F_\rho = 25\%$



(e) $F_\rho = 40\%$



(f) $F_\rho = 50\%$

Figure 23. Variation of F_ρ — (FINAL) velocity fields.

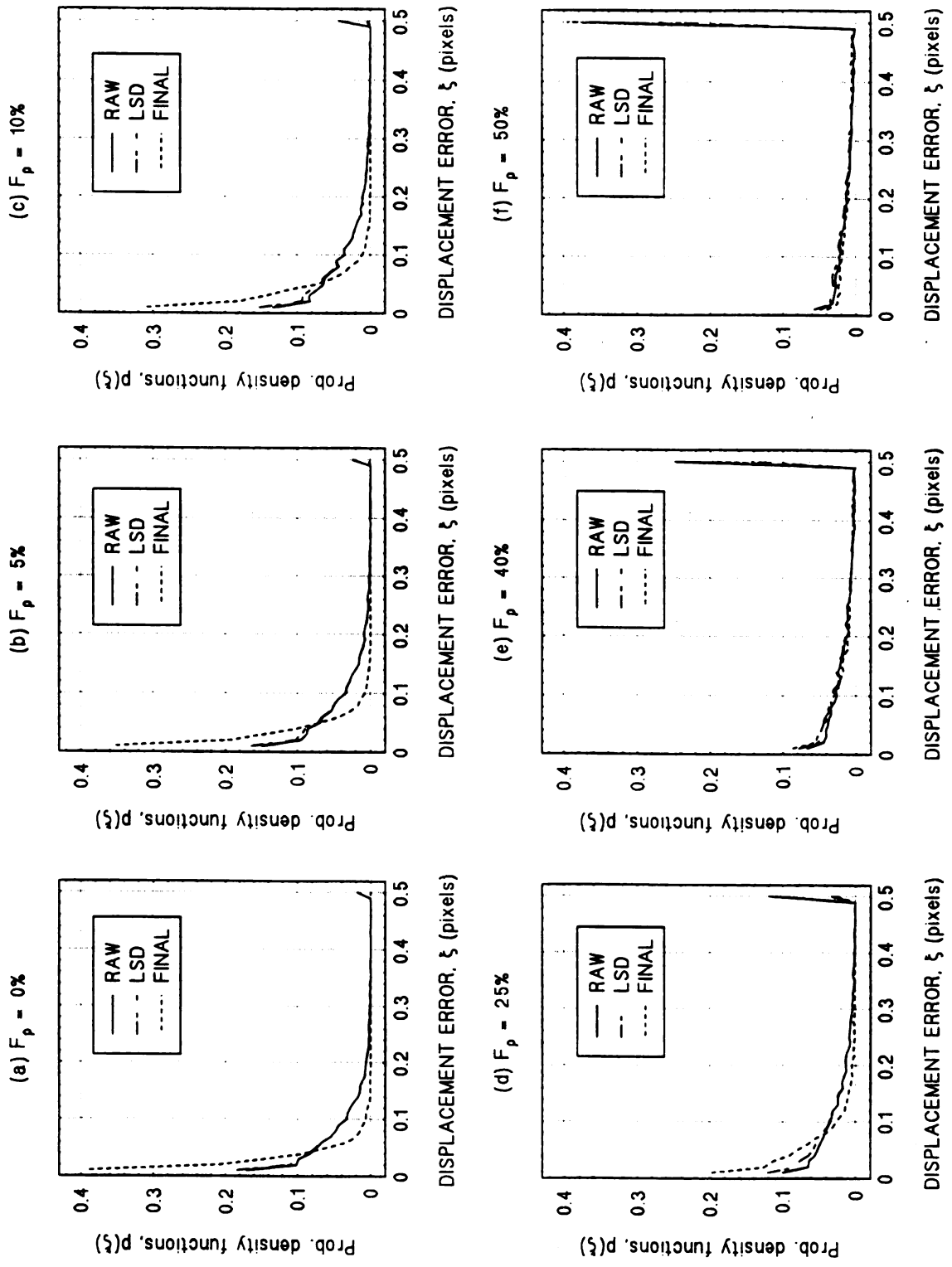


Figure 24. Variation of F_ρ — probability density functions, $p(\xi)$.

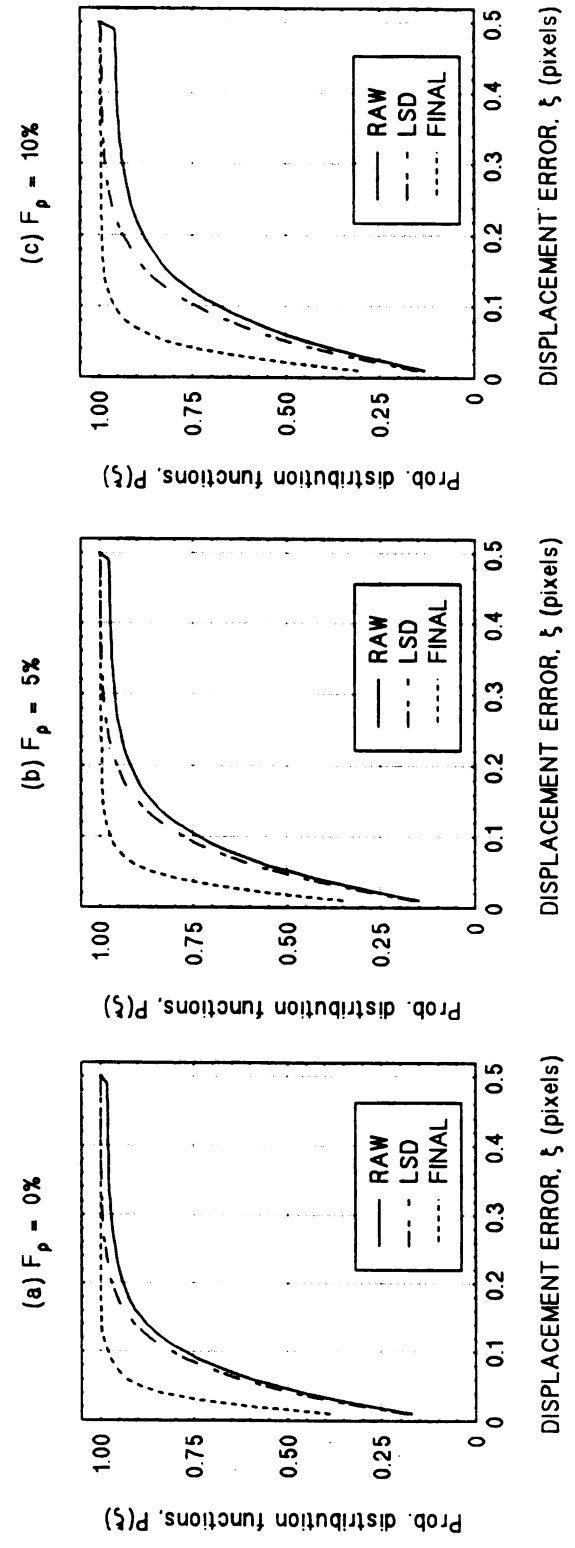


Figure 25. Variation of F_ρ — probability distribution functions, $P(\xi)$.

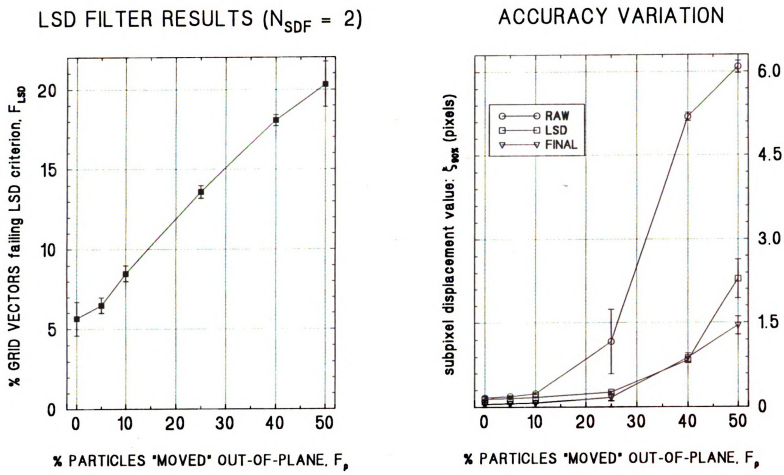


Figure 26. Variation of $F_p - \xi_{90\%}$ and F_{LSD} vs. F_p.

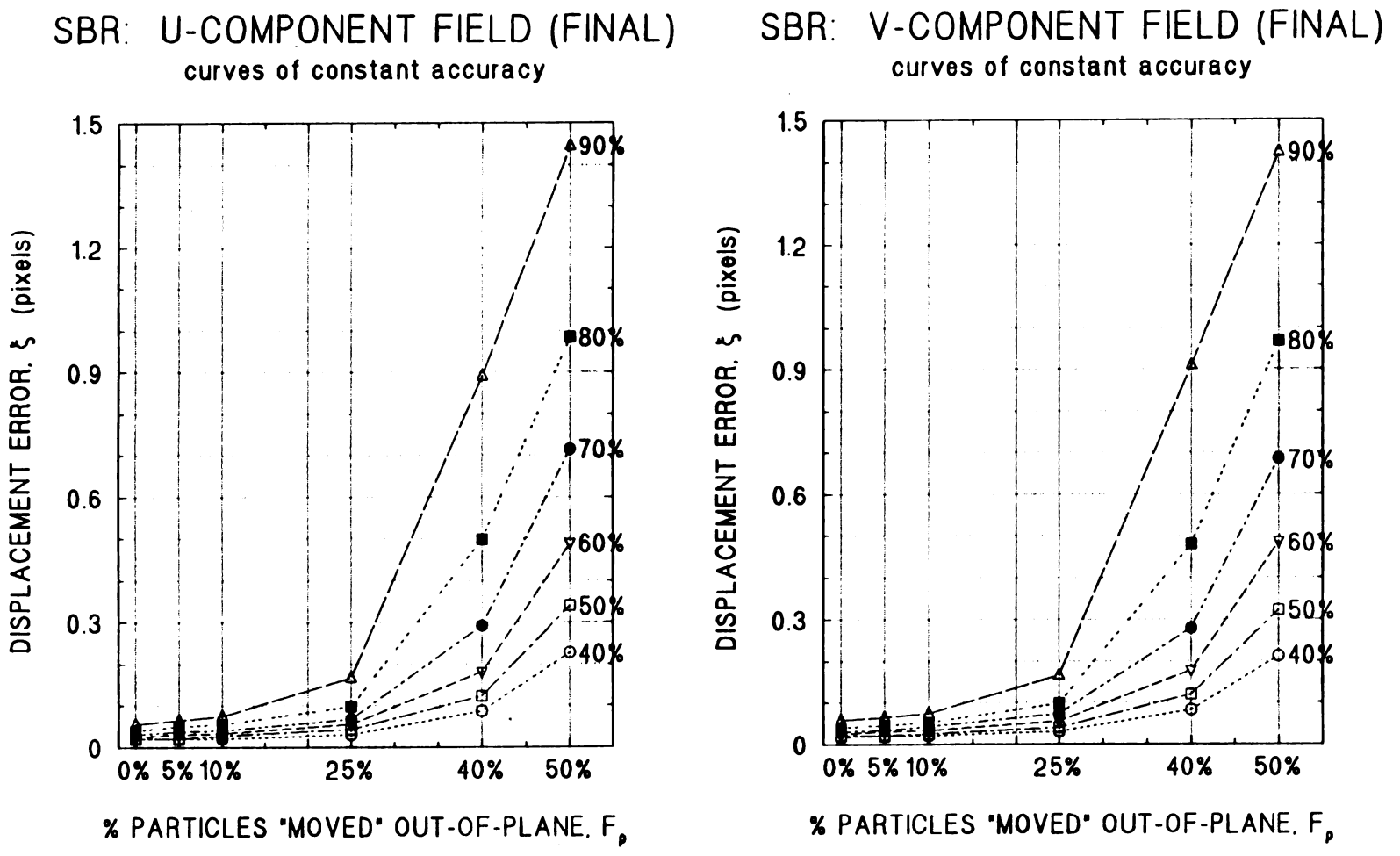


Figure 27. Variation of F_ρ — iso-accuracy plots (FINAL).

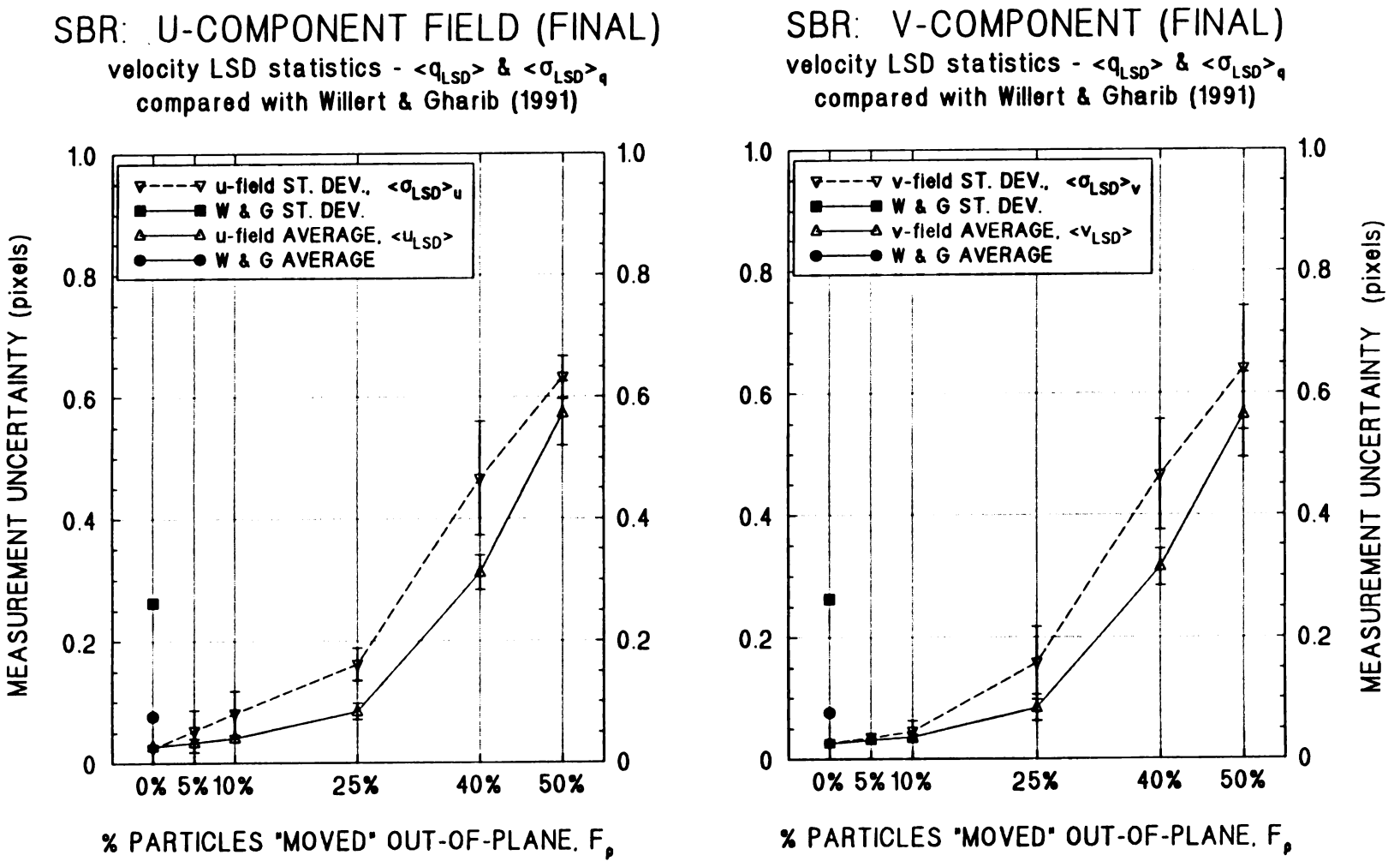


Figure 28. Variation of F_ρ — statistical uncertainty plots (FINAL).

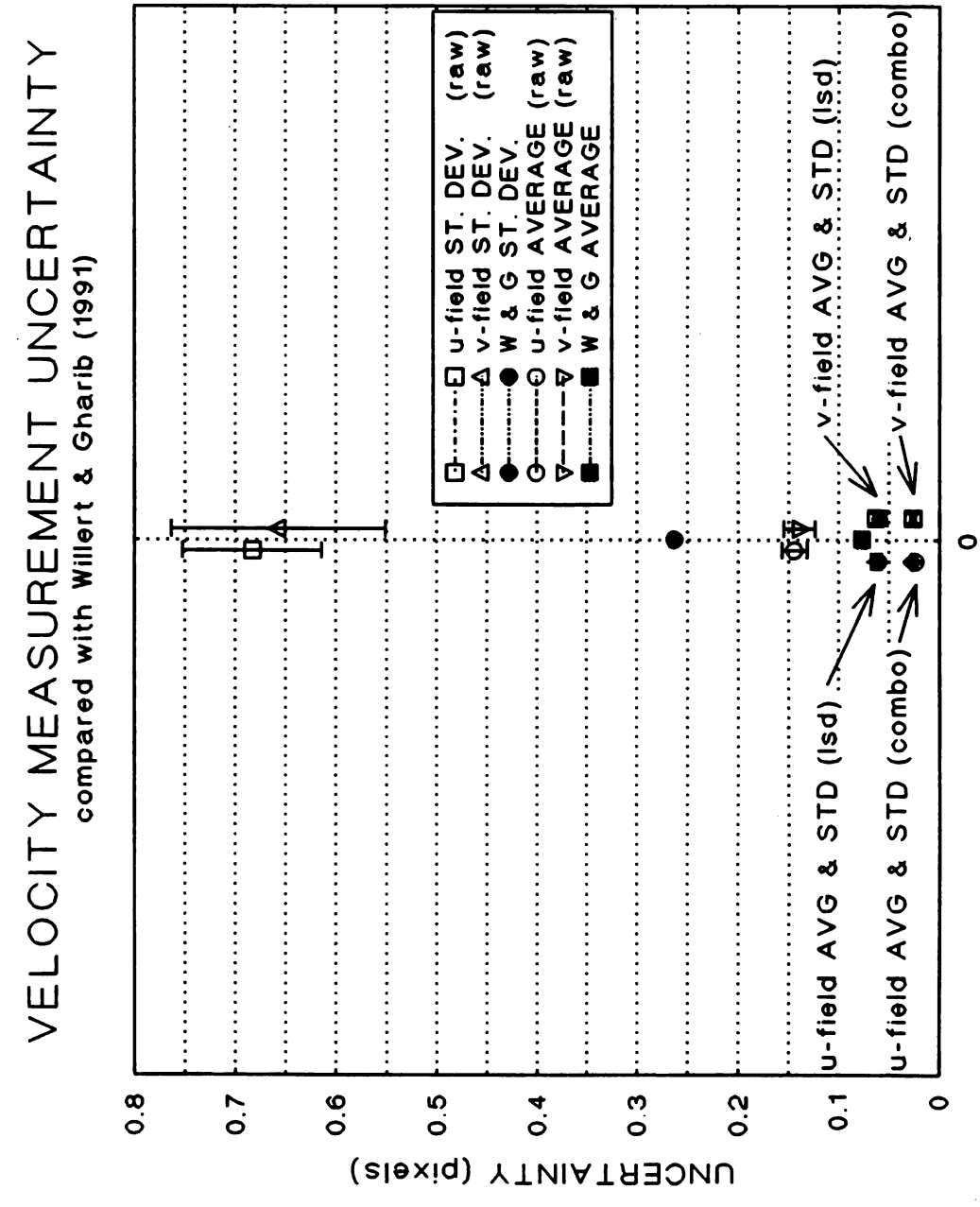


Figure 29. Variation of F_ρ — effect of filtering on statistical uncertainty.

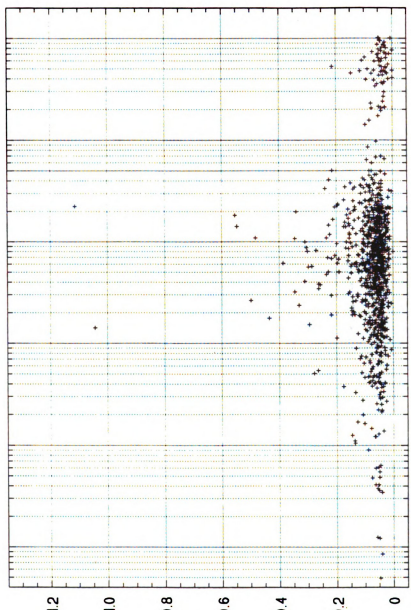


Figure 30. Randomization of I_0 [LOCAL] — (RAW) subpixel accuracy vs. w-component.

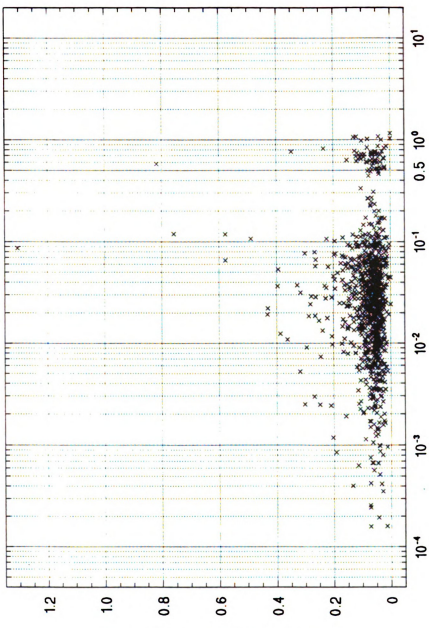


Figure 31. Randomization of I_0 [LOCAL] — (FINAL) subpixel accuracy vs. w-component.

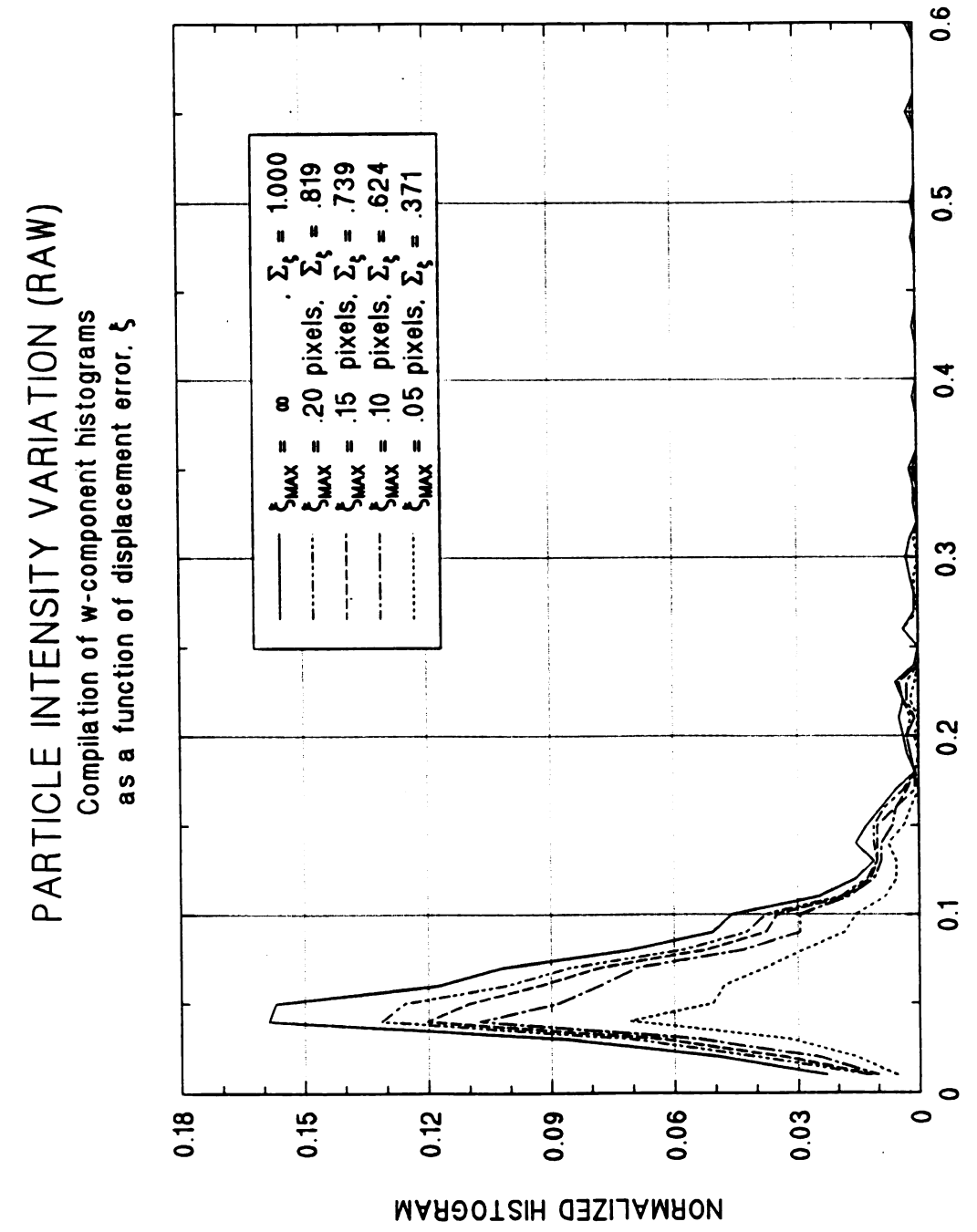


Figure 32. Randomization of I_0 [LOCAL] — histograms of "dimensionless" w-component (RAW).

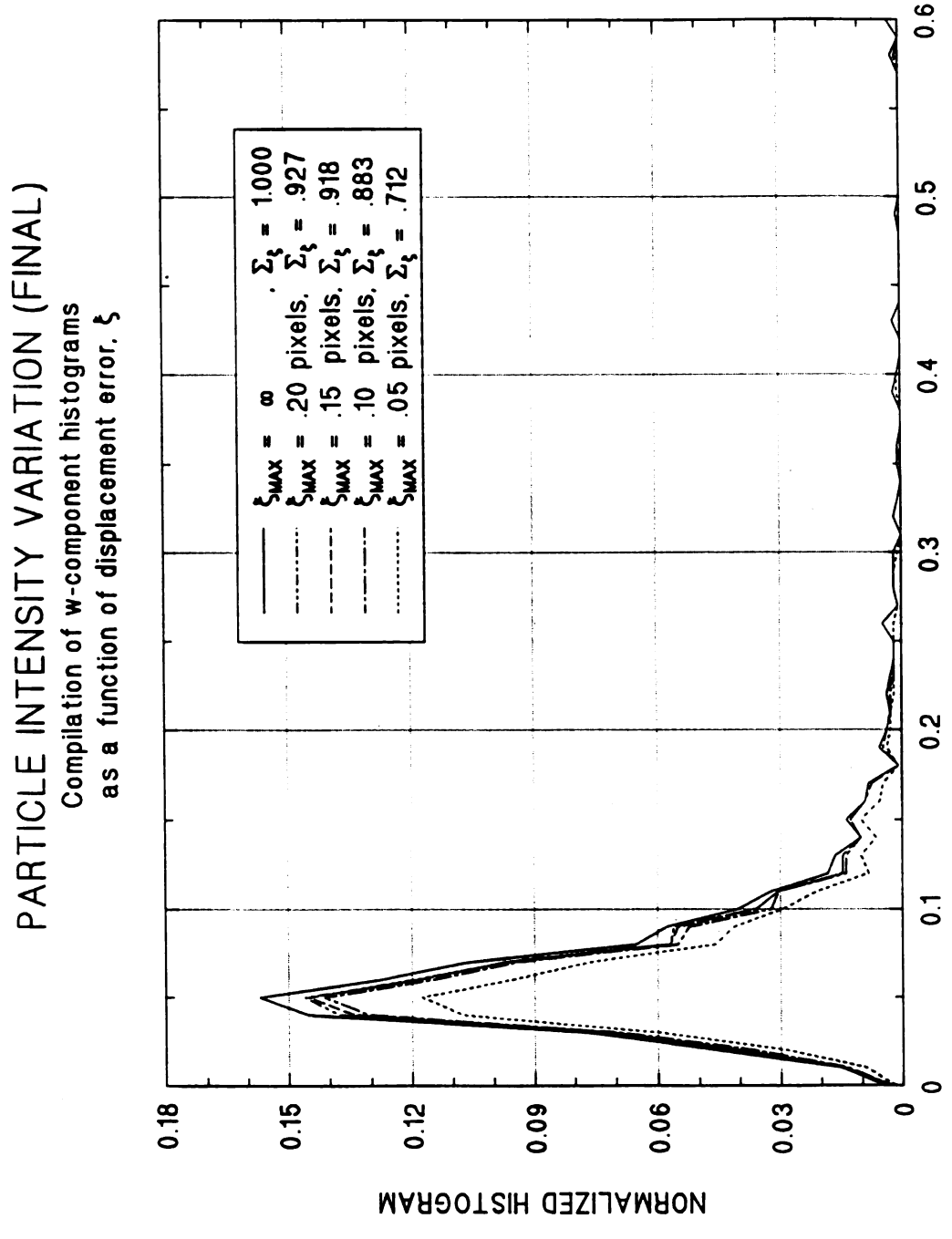


Figure 33. Randomization of I_o [LOCAL] — histograms of "dimensionless" w-component (FINAL).

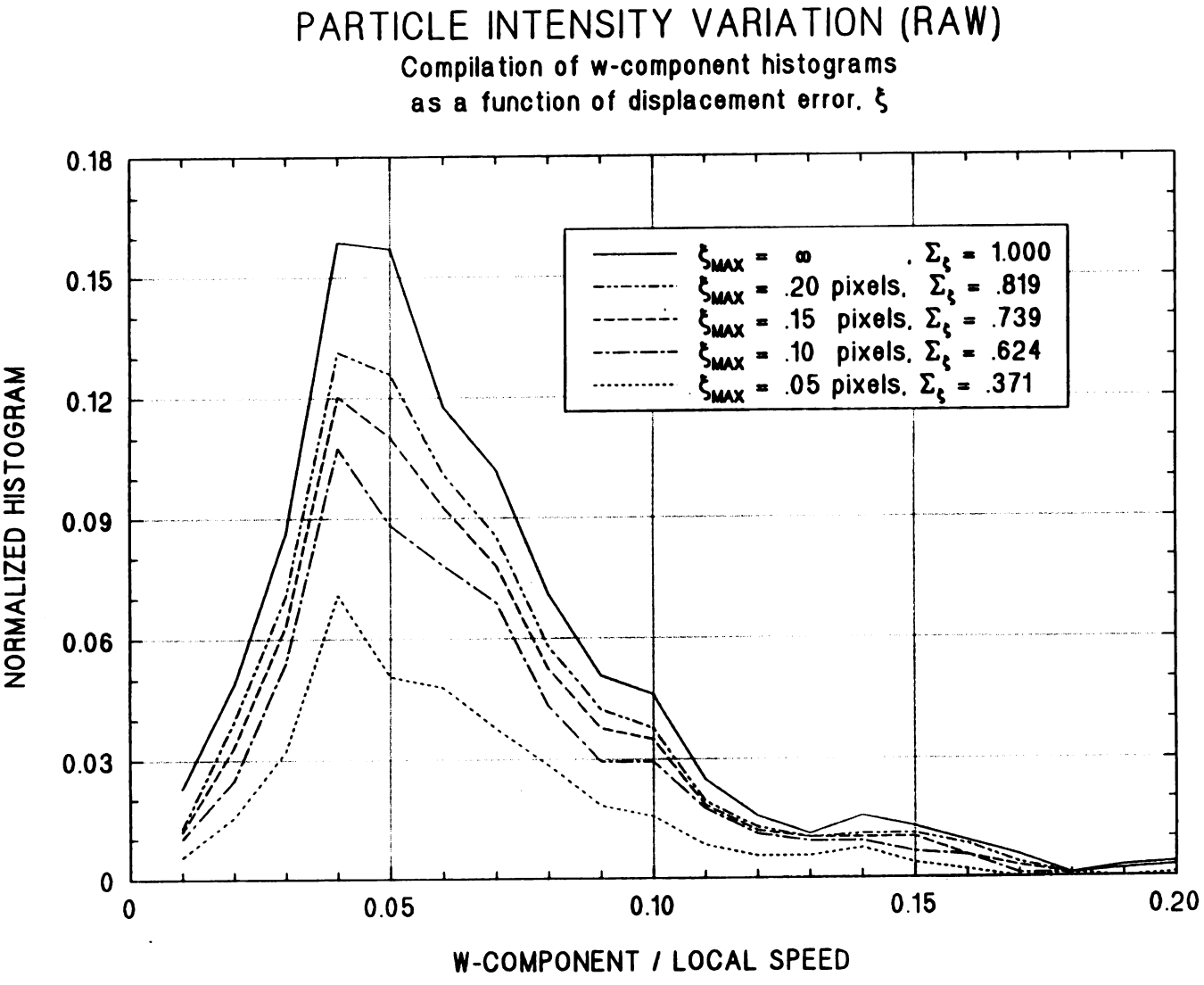


Figure 34. Randomization of I_0 [LOCAL] — histograms of "dimensionless" w-component (RAW) — scaled version.

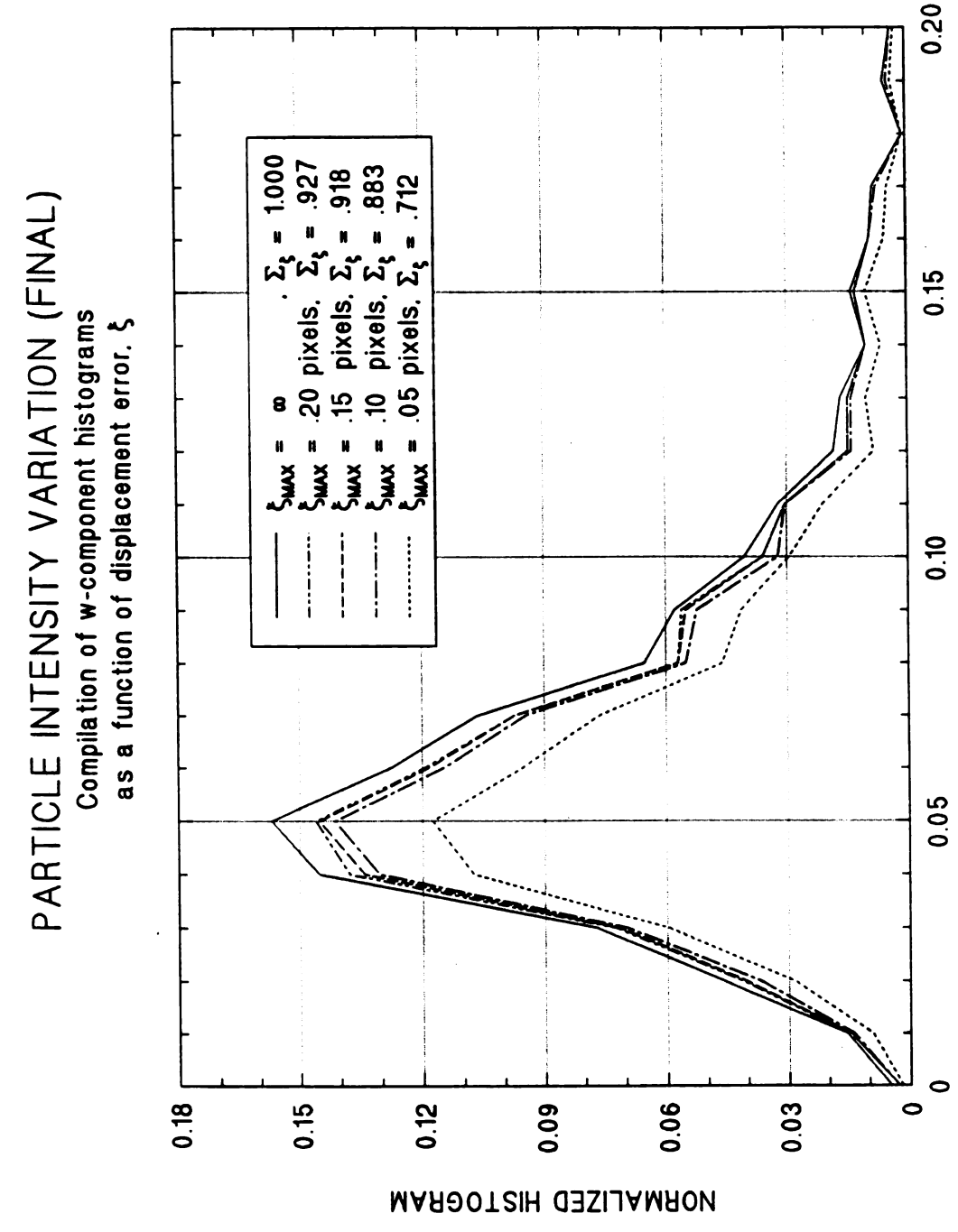


Figure 35. Randomization of I_0 [LOCAL] — histograms of "dimensionless" w-component (FINAL) — scaled version.

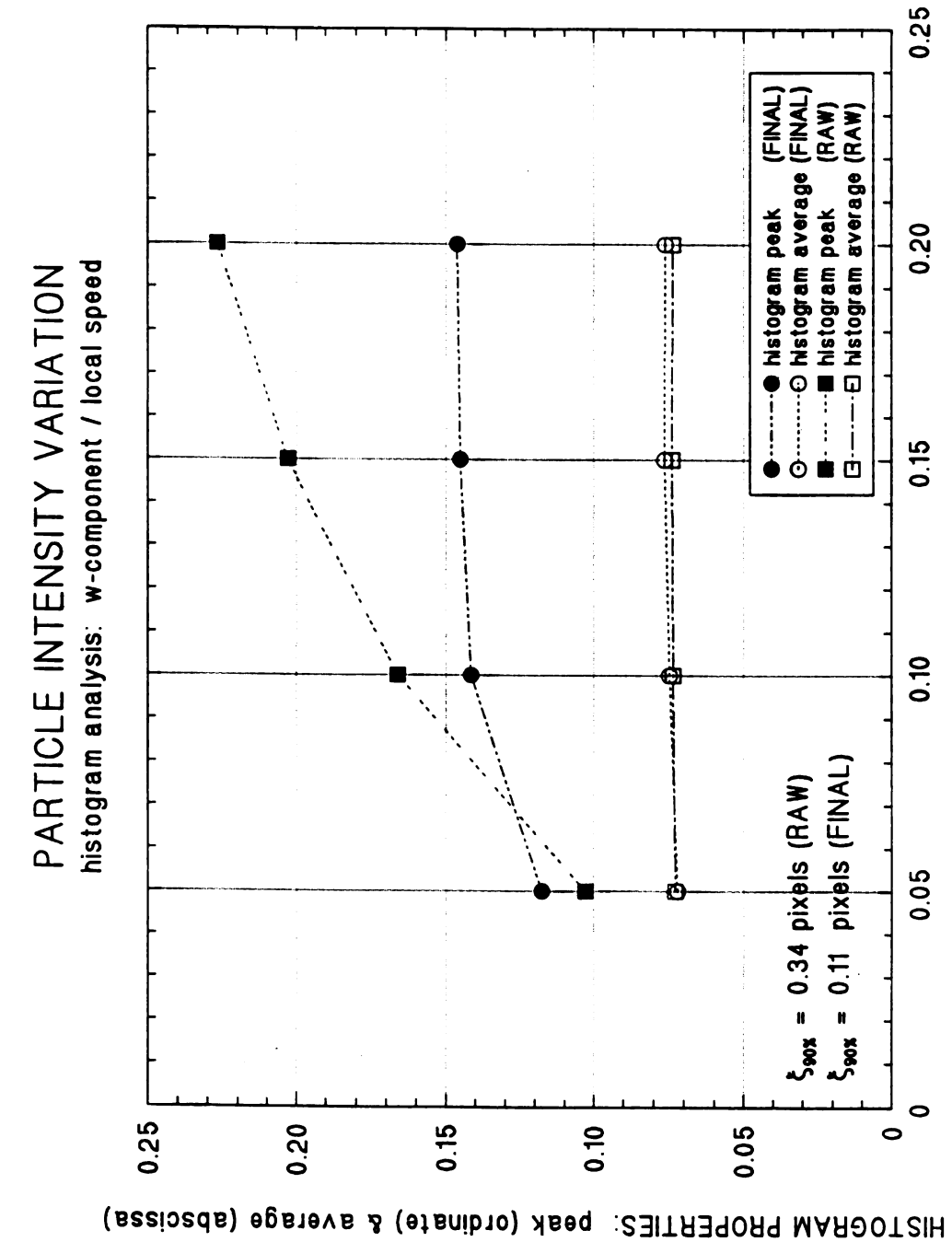


Figure 36. Randomization of I_0 [LOCAL] — behavior of histogram peaks (ordinate) and averages (abscissas).

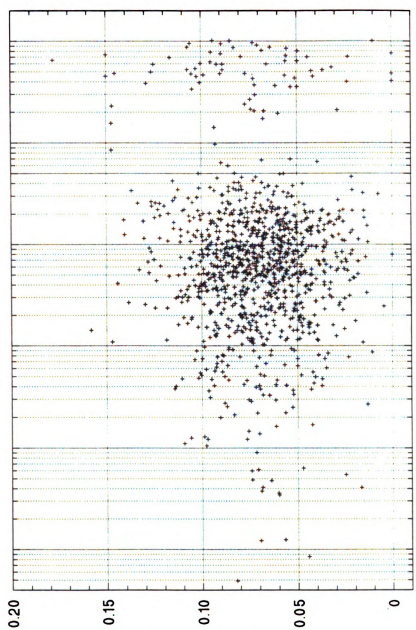


Figure 37. Randomization of I_0 [AVERAGE] — (RAW) subpixel accuracy vs. w-component.

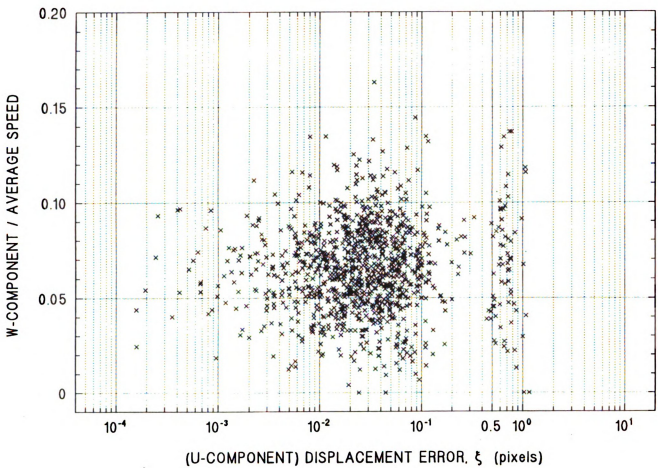


Figure 38. Randomization of I_o [AVERAGE] — (FINAL) subpixel accuracy vs. w-component.

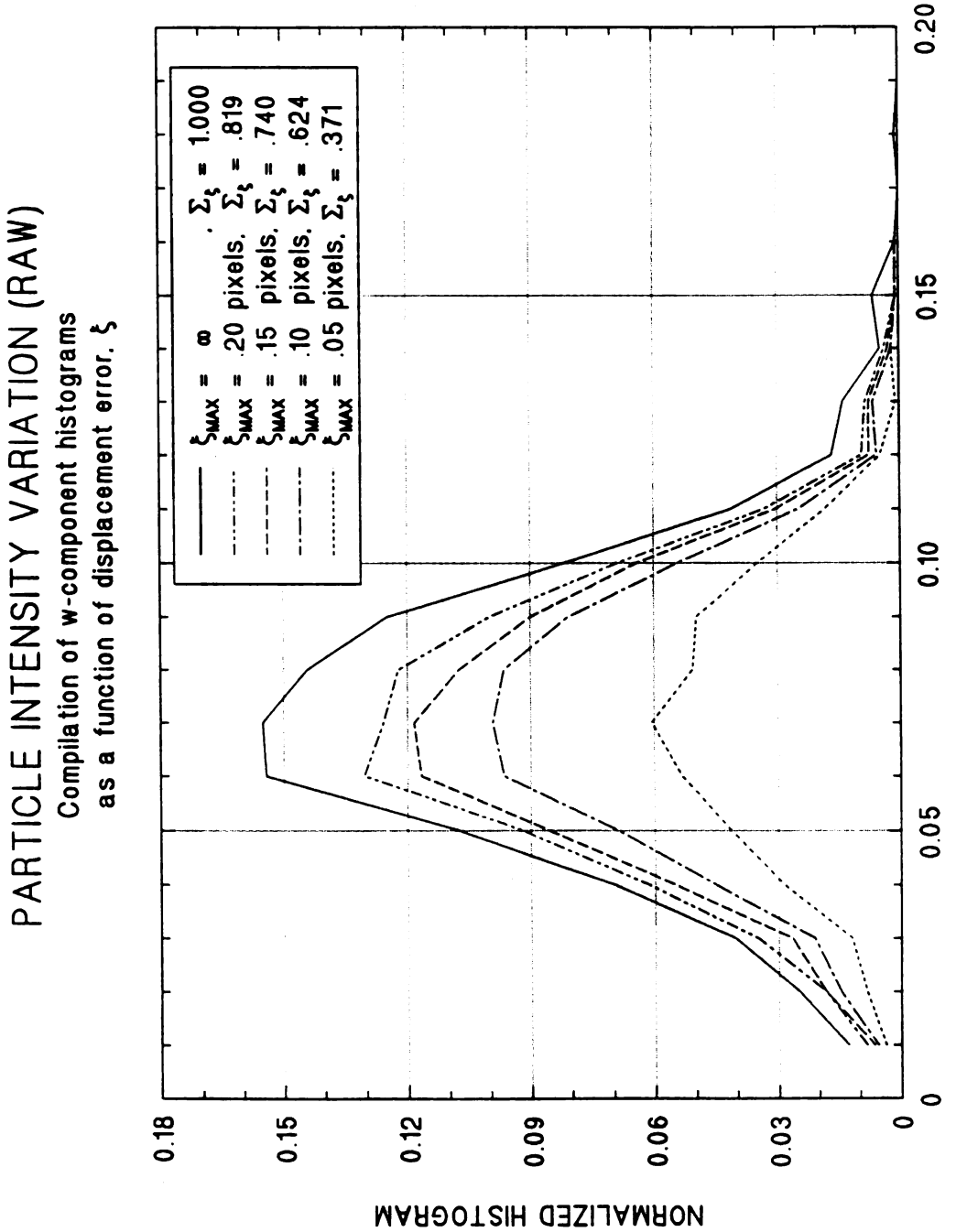


Figure 39. Randomization of I_o [AVERAGE] — histograms of "dimensionless" w-component (RAW).

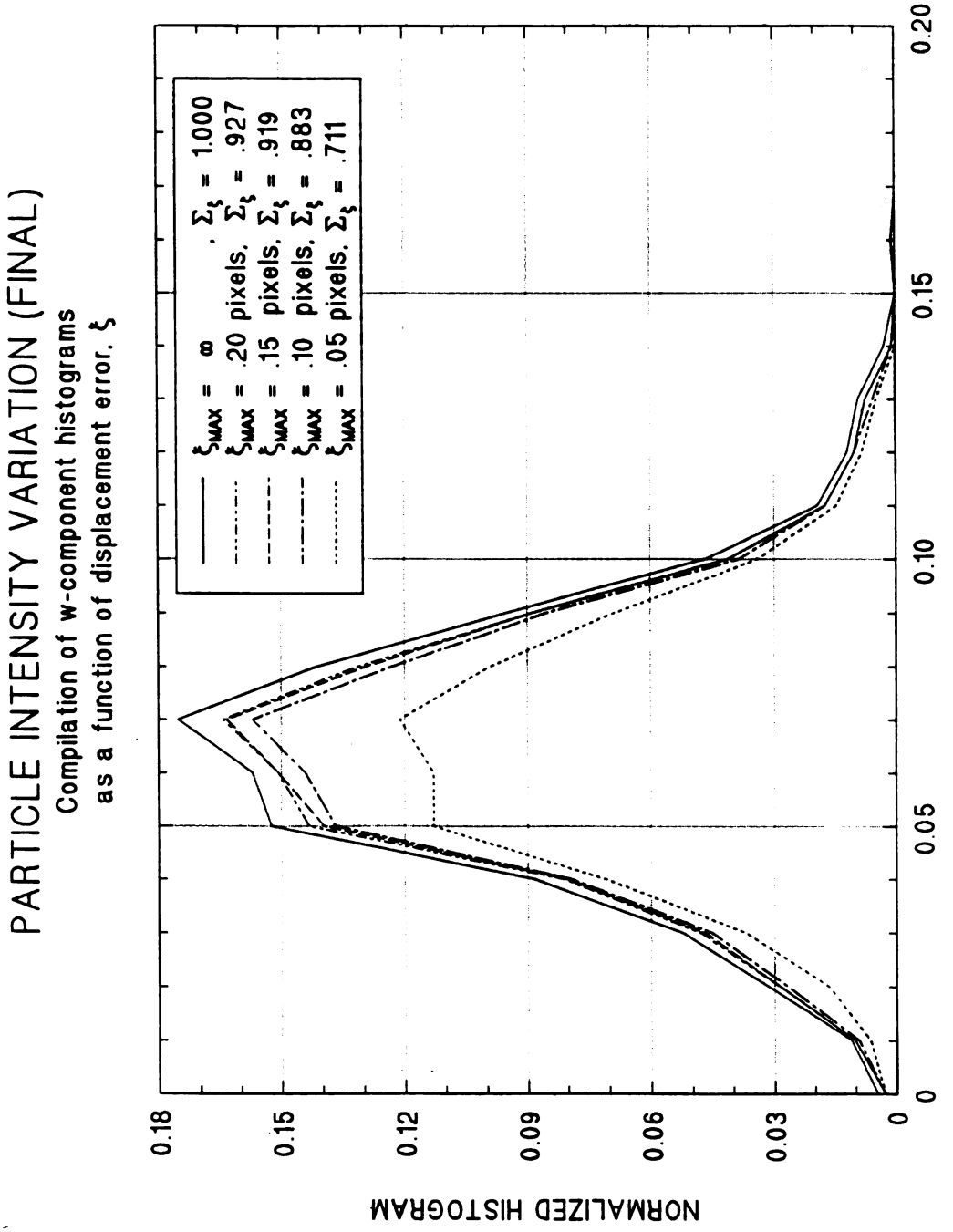


Figure 40. Randomization of I_0 [AVERAGE] — histograms of "dimensionless" w-component (FINAL).

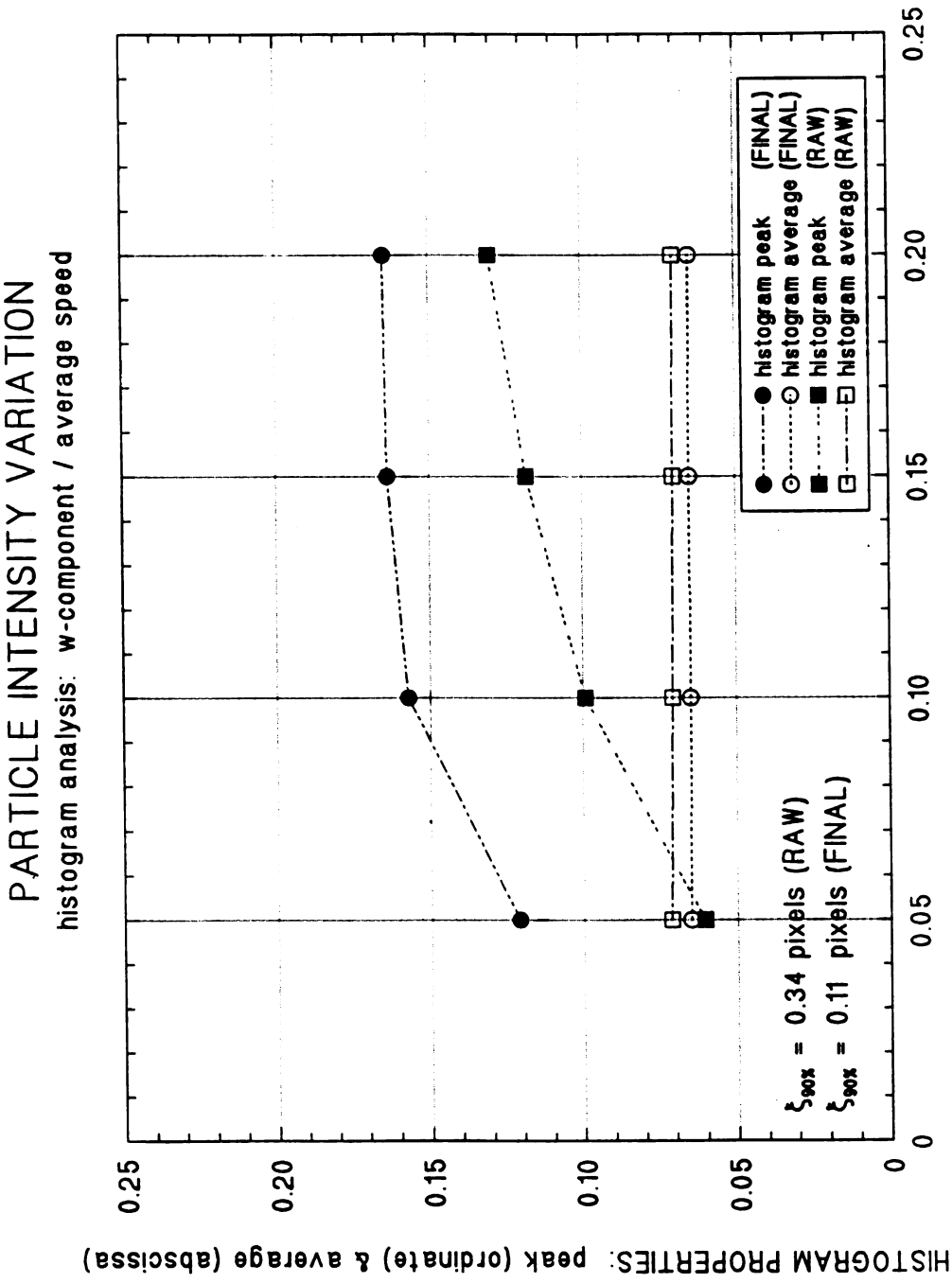


Figure 41. Randomization of I_0 [AVERAGE] — behavior of histogram peaks (ordinate) and averages (abscissas).

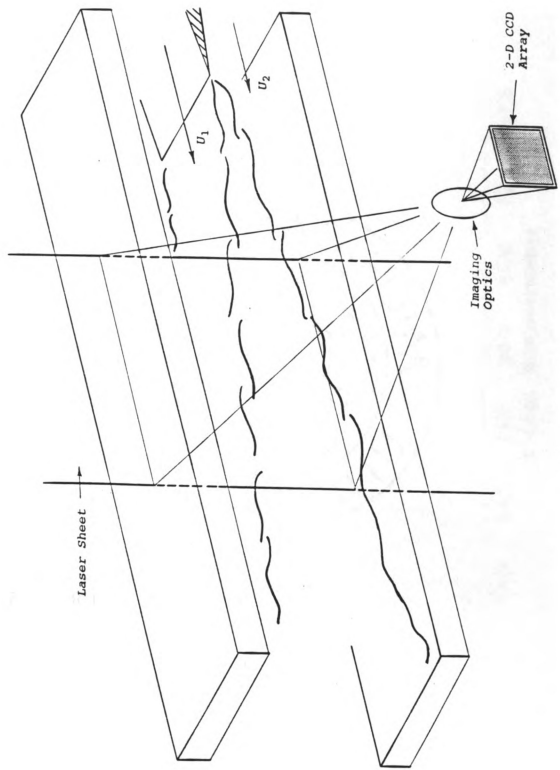


Figure 42. Wake-shear layer flow — experimental setup.

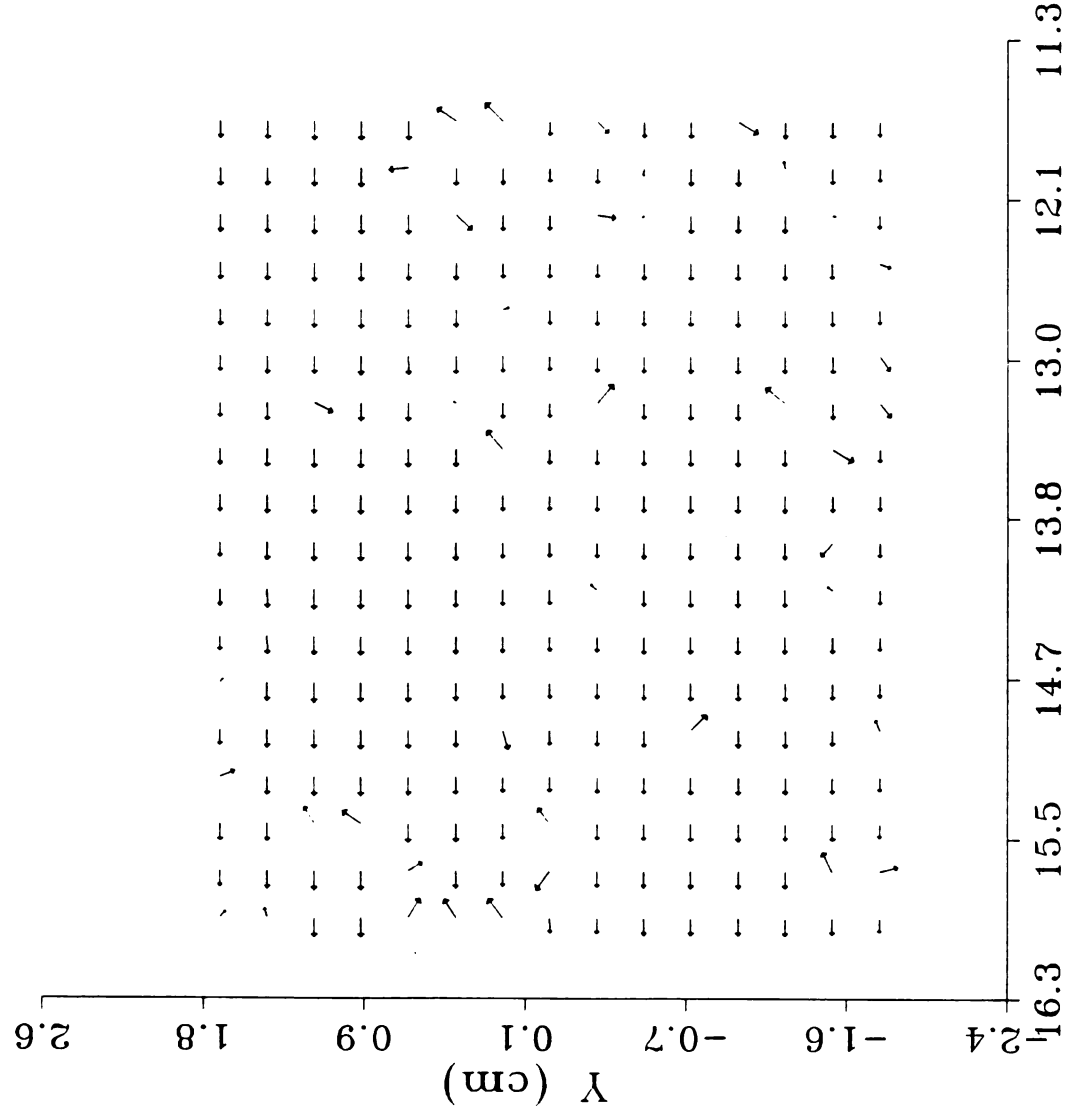


Figure 43. Wake-shear layer flow — (RAW) velocity field.

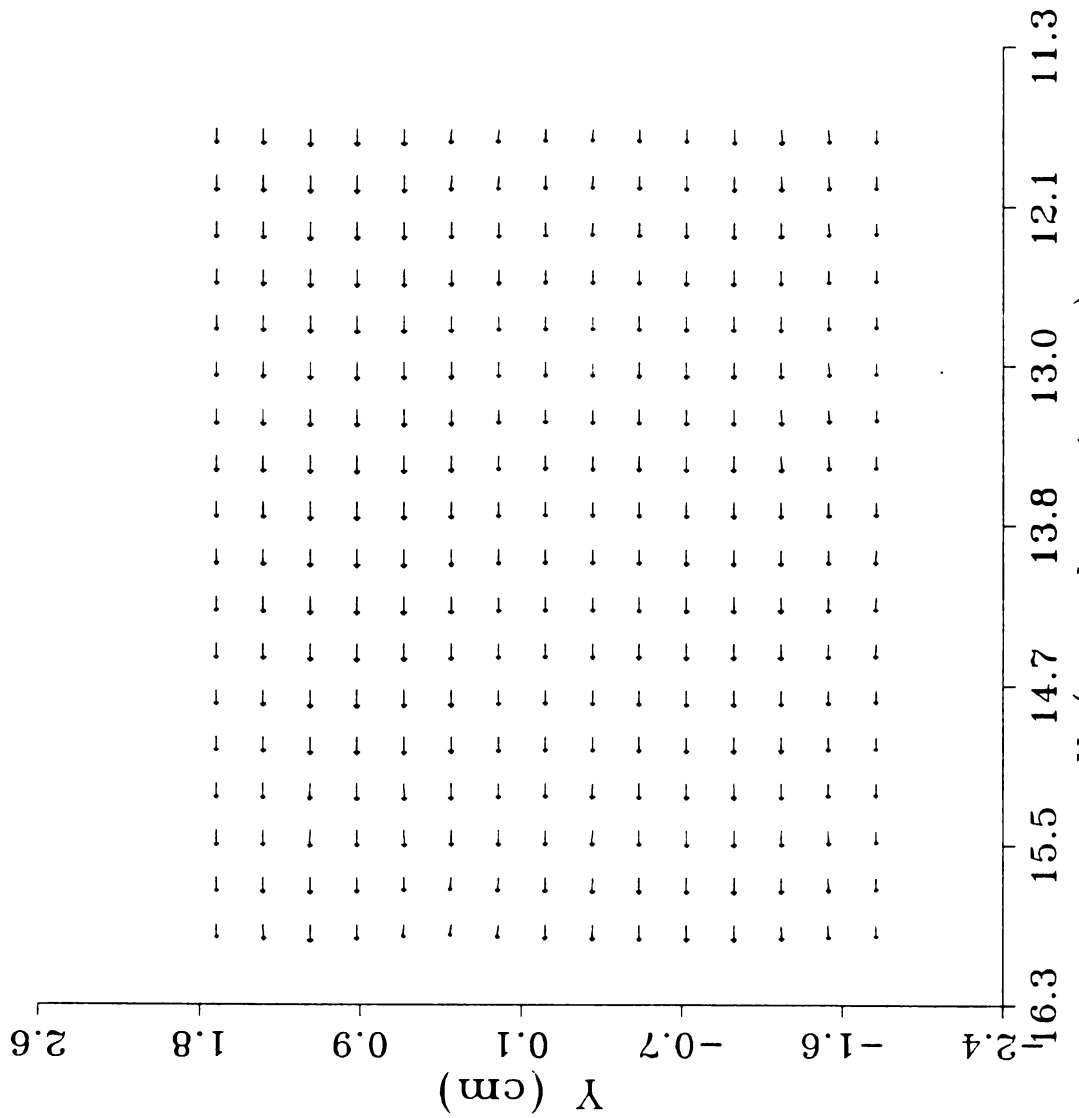


Figure 44. Wake-shear layer flow — (FINAL) velocity field.

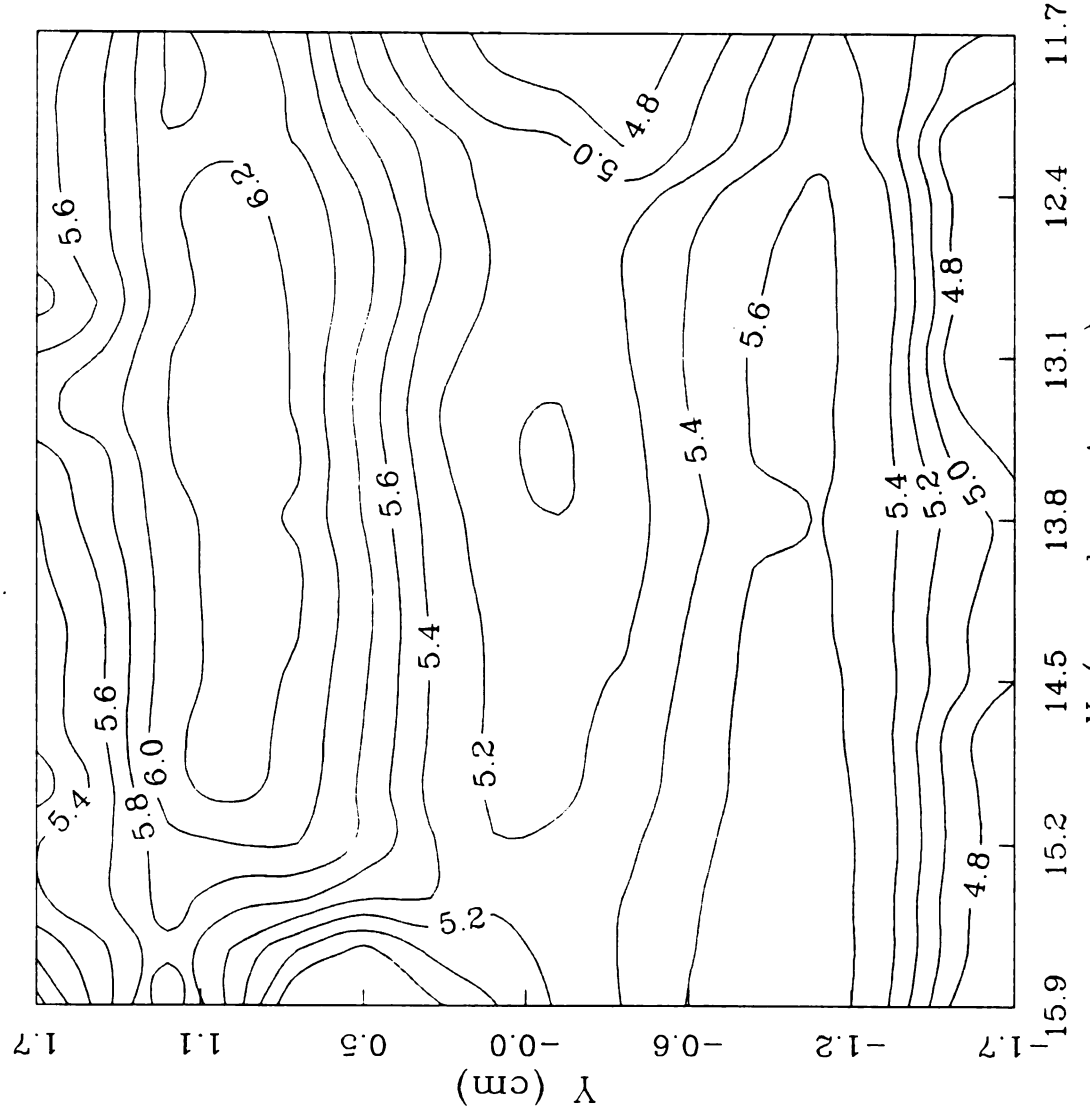


Figure 45. Wake-shear layer flow — (FINAL) velocity contour field.

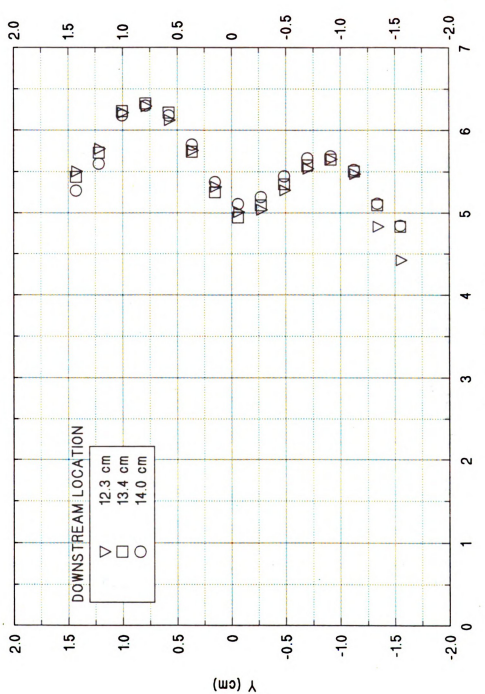


Figure 46. Wake-shear layer flow — (FINAL) velocity profiles.

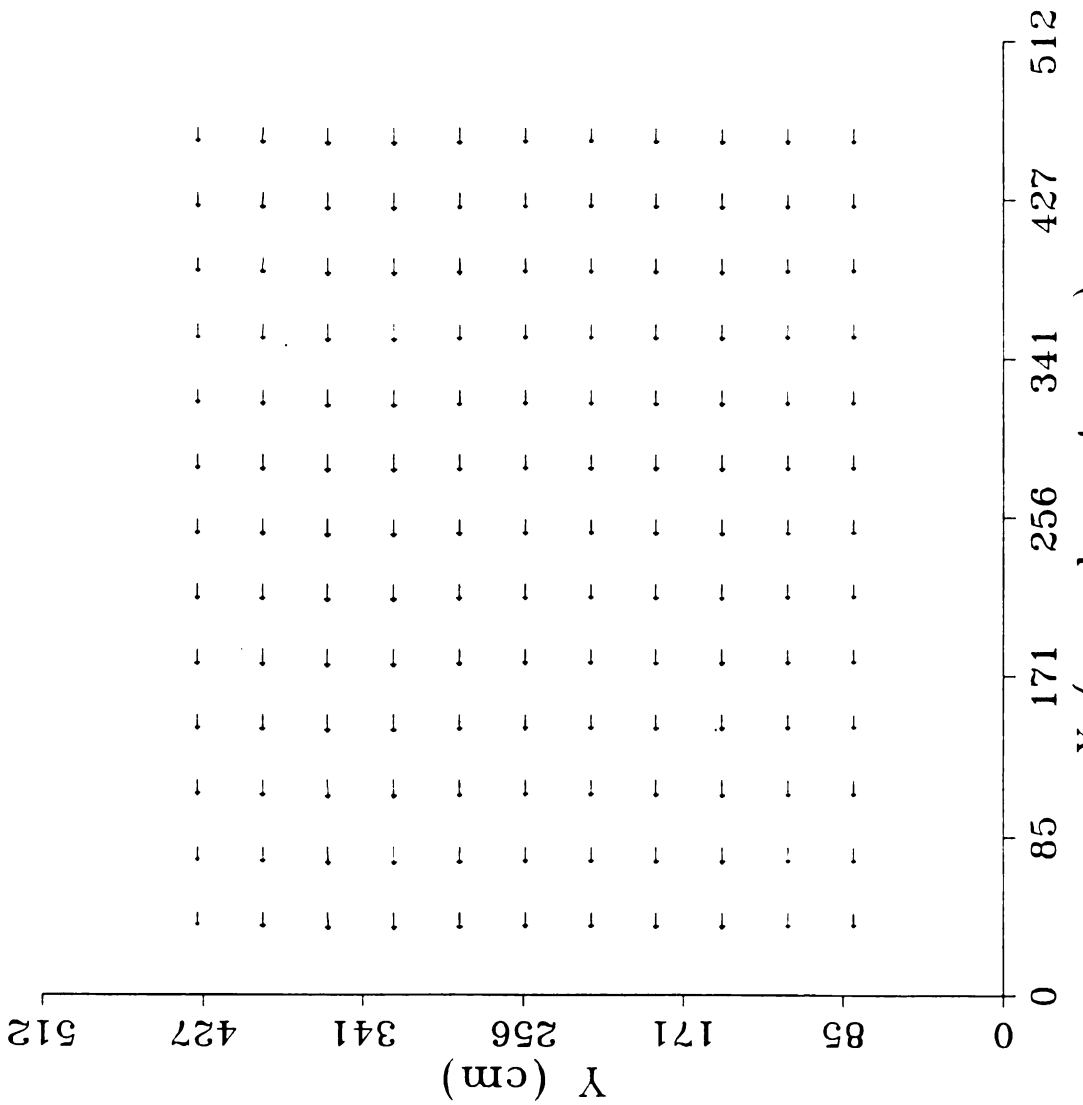


Figure 47. Wake-shear layer flow — (FINAL) velocity field. [interpolated field images]

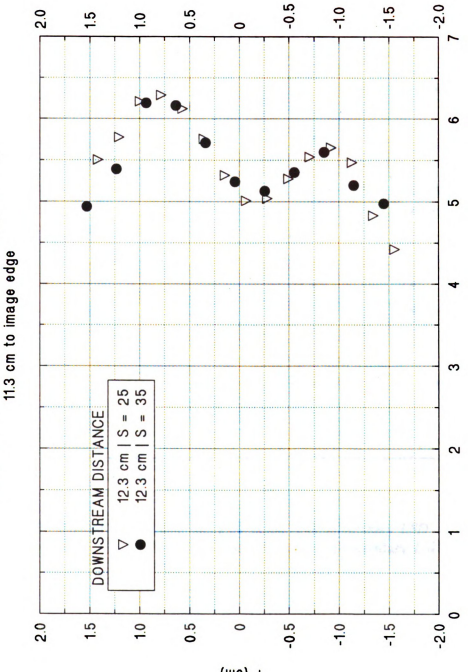


Figure 48. Wake-shear layer flow — comparison of (FINAL) velocity profiles.

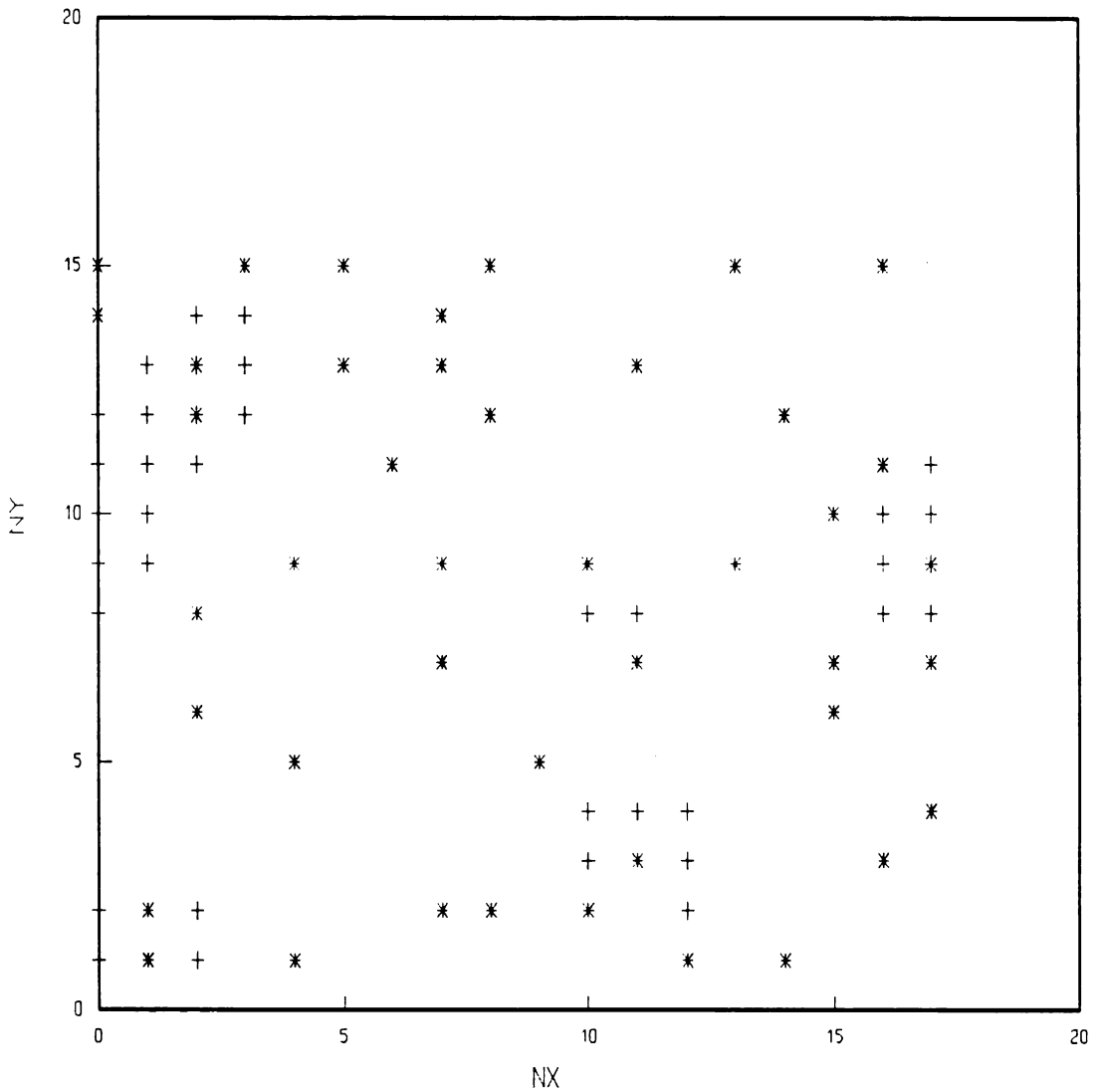


Figure 49. Spatial distribution of: grid points failing primary *LSD* criterion (*); grid points failing secondary *LSD* criterion (+).

APPENDICES

APPENDIX A: Sample window grid

A.1. Edge effects

If the grid coordinate system is defined such that a sample window is too close to the edge of the image, it is possible that the corresponding (concentric) roam window would be defined such that it is partially off the image. This would happen if

$$\frac{R-1}{2} > m + \frac{S-1}{2} \qquad \text{(A-1)}$$

with m as defined by equation (4). This circumstance was corrected by increasing m to render the above inequality always untrue. This modification is calculated as follows:

$$m_i = \frac{MOD(512,S)}{2} \quad ; \quad m_i = \textit{initial margin} \qquad \text{(A-2)}$$

This result is then compared to the maximum resolvable displacement, which is obtained from half the size difference between roam and sample windows,

$$\Delta_{MAX} = \frac{R-S}{2} \qquad \text{(A-3)}$$

If $m_i < \Delta_{MAX}$, then the image margin is *increased* to exactly match that same displacement. This increase in turn implies a reduction in the number of grid points per row, n. Therefore, n must be recalculated for the new image margin, m_0. The resulting remainder from *that* calculation is then halved and added to the maximum displacement to get the final image margin. The previous process expressed in equation form is,

$$m_0 = \frac{R - S}{2}$$

$$n = \frac{512 - 2m_0}{2} \qquad \text{(A-4)}$$

$$m = m_0 + \frac{MOD[(512 - 2m_0), s]}{2}$$

This is the actual image margin used. Despite this increased value, it is sometimes possible that the edge sample windows could be close to the test section walls. As a result, these windows would encompass near-wall boundary layers, ie., regions of relatively high velocity gradients (an error source, as stated in §2.1). In this event, the user has the option of eliminating entire grid rows or columns from consideration. This typically requires an estimate of the size of the high gradient region.

A.2. Window coordinate systems

The coordinates of the initial sample window center point in units of pixels are

$$x_{c0} = m + \frac{S - 1}{2}$$
$$y_{c0} = m + \frac{S - 1}{2} \qquad \text{(A-5)}$$

If n_x and n_y represent the horizontal and vertical coordinates of a sample window on the [0,n-1] grid scale of Figure 4, then the conversion from grid scaling to pixel scaling is

$$\begin{aligned} x_c(n_x) &= x_{c0} + n_x S \\ y_c(n_y) &= y_{c0} + n_y S \end{aligned} \quad ; \quad n_x, n_y \in [0, n-1] \qquad \text{(A-6)}$$

APPENDIX B: Evaluation of correlation peak

The subpixel peak location is determined from the relative maximum of equation (13). This is accomplished by taking the natural logarithm of both sides and differentiating once each with respect to x and y:

$$\frac{\partial(\ln F)}{\partial x} = C_1 x + 2C_2 x$$
$$\frac{\partial(\ln F)}{\partial y} = C_3 y + 2C_4 y \qquad \text{(B-1)}$$

Setting the derivatives equal to zero and solving yields the peak's coordinates relative to the center of the 3 × 3 pixel^2 area:

$$(x_{SUB}, y_{SUB}) = \left(-\frac{C_1}{2C_2}, -\frac{C_3}{2C_4} \right) \qquad \text{(B-2)}$$

Due to the size of the area being fitted, these results should be less than 1 pixel. This criterion fails if the nature of the peak is not optimal. For example, the magnitude of the peak could be too close to the background level of the correlation domain. Or the peak could be truncated because it is too close to the edge of the domain. Peak truncation can be mitigated by fitting around an interior point adjacent to the peak, rather than the peak itself. This corrective measure will work if, in fact, the peak is in between the edge and interior locations; ideally, a subpixel fit around either one would reveal the true location. If a measurement nevertheless yields a subpixel adjustment greater than 1 pixel, then the result is designated as a bad fit and the (n_x,n_y) grid location of that

particular sample window is stored. The subpixel adjustments are set to zero, and the original estimate to the nearest pixel is taken as the best available result. Otherwise, the adjustment is added to the nearest pixel estimate to produce the peak location (x_p,y_p).

The user has a final option to invalidate this measurement by setting a *correlation peak threshold* as one of the initial parameters in the velocity field processor. If the peak value is not greater than this threshold, then the velocity will be set to exactly zero. This is sometimes useful as a diagnostic to determine what regions of the flow produced the best correlations for their velocity measurements.

APPENDIX C: Determination of $\Gamma(r)$

This details the calculation of $\Gamma(r)$ referred to in §2.4.2, involving the line integration of velocity around a quarter-circular contour. The current implementation is based on a fixed origin corresponding to that of the solid body rotation flow examined in §3. The capability for a variable origin is readily available through some modifications in the integration routines in the software. At this point, the flows being investigated have not necessitated these modifications; therefore the option is retained for future experimentation.

The description here continues from equation (33), which subdivided the contour into horizontal, vertical, and curved sections denoted Γ_x, Γ_y, and Γ_c respectively. The first two terms are easiest to evaluate, because only one component of velocity contributes to the integral along each of these paths. First, the path-averaged velocity components were determined:*

$$\bar{v}_y = \sum_{j=0}^{r} \frac{v(0,j)}{r+1} \quad , \quad \bar{u}_x = \sum_{i=0}^{r} \frac{u(i,0)}{r+1} \qquad \text{(C-1)}$$

The subscripts on v_y and u_x emphasize the spatial direction over which the average is computed. The circulation along these contour sections arises simply from multiplication of these averages with the total length of the paths, ΔL, which is simply the radius being examined (although ΔL is not, in fact, equal to r because the two are defined in different

*See footnote on page 44 for explanation on units of r.

coordinate systems (and units); ΔL is in pixels, r is in the (n_x,n_y) grid coordinate space). The expressions for circulation are therefore,

$$\Gamma_y = -\bar{v}_y \Delta L \quad , \quad \Gamma_x = \bar{u}_x \Delta L \qquad \text{(C-2)}$$

A last characteristic of this issue is that *u(y)* and *v(x)* are ideally uniform in x and y respectively for solid body rotation. This implies that equation (C-2) is probably adequate, since the functions are generally well-behaved; even the measured velocity fields mostly show minor deviations in *u(y)* and *v(x)*.

The curved section of the contour integral is less trivial because both velocity components must be estimated. Also, the curve itself must be approximated as a continuous series of line segments (cf. Figure 7). Note that this is a rather poor approximation at extremely low radii, since the number of segments approximating the curve is small.

The task at this point is to use this approximation to estimate the average velocity along each segment. Consider Figure C.1, which shows a typical initial segment of a contour's curved portion. The initial velocity at point 1 is already available from the data, as are the velocities at points 3 and 4. Spatial interpolation of the latter two points yields the velocity at point 2. Another spatial interpolation of points 1 and 2 then yields the average u-/v-components along the line segment. The contribution to the line integral from this segment of the circular arc becomes,

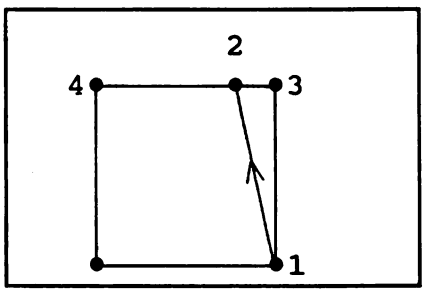


Figure C.1. *Γ(r)* integration contour segment (1-2).

$$\Gamma_c = \oint \vec{u} \cdot d\vec{l} = \bar{u}\Delta x + \bar{v}\Delta y \qquad \text{(C-3)}$$

where Δx and Δy would represent the horizontal and vertical projections of the circular arc line segment. In Figure C.1, the line segment 1-2 has Δx and Δy equal to the distances between points 2-3 and 1-3 respectively.

The final point of this segment (2) becomes the initial point of the next segment; an analogous two-step interpolation process will then yield the velocity of the final point of *that* segment, and so on. Note that the only requirement here is a geometric definition of the curve (which corresponds to Figure 7 in this circumstance). If necessary, a more arbitrary curve (than a circle) could be defined without needing drastic changes in the interpolation and integration process.

REFERENCES

REFERENCES

ADRIAN R J (1986) Multi-optical measurements of simultaneous vectors in unsteady flow —a review *Int. J. Heat and Fluid Flow* **7** 127-145

ADRIAN R J (1988) Statistical properties of particle image velocimetry measurements in turbulent flow *Laser Anemometry in Fluid Mechanics* **Vol. III** (LADOAN Institute Superior Tecnico, Lisbon, Portugal) 115-129

ADRIAN R J (1991) Particle-imaging techniques for experimental fluid mechanics *Annu. Rev. Fluid. Mech.* **23** 261-304

FINCHAM A, BLACKWELDER R, SPEDDING G (1991) Current Constraints of Digital Particle Tracking in Fluid Flows *Amer. Phy. Soc.* **Vol. 36, No. 10** 2692

KEANE R, ADRIAN R J (1990) Optimization of particle image velocimeters. Part I: double pulsed systems *Meas. Sci. Technol.* **1** 1202-1215

LOURENCO L M, KROTHAPALLI A (1987) The role of photographic parameters in laser speckle or particle image displacement velocimetry *Exp. Fluids* **5** 29-32

LOURENCO L M, KROTHAPALLI A, SMITH C A (1989) Particle Image Velocimetry *Advances in Fluid Mechanics Measurements* (Lecture Notes in Engineering 45, Ed. M Gad-el-Hak Springer-Verlag) 127-199

PRESS W H, FLANNERY B P, TEUKOLSKY S A, VETTERLING W T (1986) Numerical Recipes: The Art of Scientific Computing *Cambridge University Press* 29-38, 509-528

ROSENFELD A, KAK A C (1982) Digital picture processing. Vol. 1 & 2 *Academic Press*

WILLERT C E, GHARIB M (1991) Digital particle image velocimetry *Exp. Fluids* **10** 181-193