A Model based Validation strategy for Cross Correlation PIV

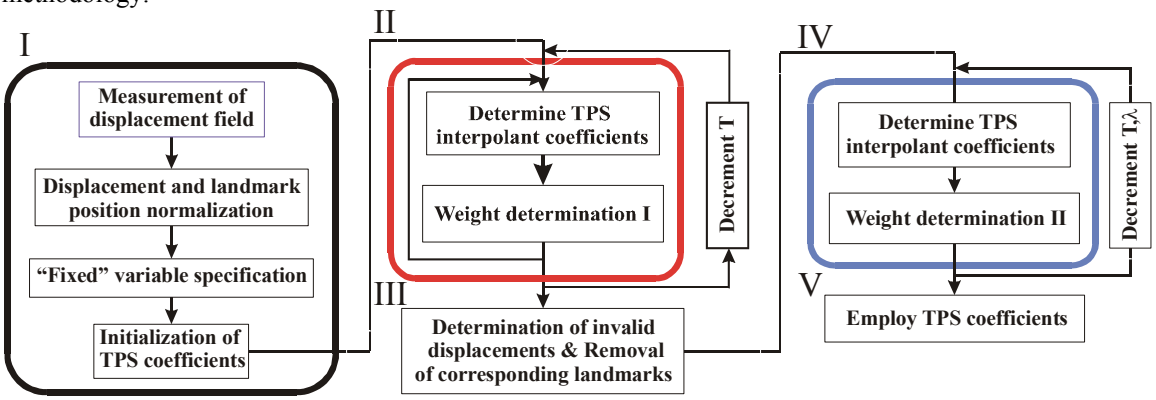
C.N. Young[1], D.A. Johnson[2], E.J. Weckman[3]

Department of Mechanical Engineering
University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
[1] c2young@uwaterloo.ca [2] da3johns@uwaterloo.ca [3] eweckman@uwaterloo.ca

Abstract

One of the major difficulties with standard cross-correlation PIV is the inherent link between the density of vectors in the measurement field and the maximum measurable displacement. Advanced hierarchical/multi-resolution/multi-pass strategies are designed to reduce this displacement/resolution link. Sparsely populated displacement fields obtained on a first pass through the image field are assumed to capture the large-scale structure of the flow field. Subsequent passes employ this large-scale information to determine the small-scale structure of the flow field. Hierarchical methods can be effective if inter-iteration results can be compensated for the presence of "false" or "invalid" vectors that contaminate the displacement field. Common compensation/validation schemes are based on a first order difference or nearest neighbour similarity constraint. Unfortunately these strategies are not entirely effective when substantial velocity gradients exist in the flow field. Difficulties can be compounded by the fact that the cross-correlation algorithm is most apt to return invalid vectors in high gradient regions.

In this paper the design of a more advanced validation framework is described. It is designed to simultaneously take account of 1) the degree of smoothness in the displacement field, 2) the similarity between particle local image properties, and 3) the degree of certainty about raw displacement measurements. The methodology can be used with vector positioning schemes (Young et al. (2002)) where displacement vectors are not arranged on a regular Cartesian grid. The algorithm, shown below, is based on the implementation of a regularized thin plate spline model (TPS) and employs an alternating update methodology.



After the measurement of the raw displacement field, a number of control parameters are initialized and the regularized displacement field is calculated using a linear least squares formulation. This regularized displacement field is used to initialize an iterative "*relaxation*" scheme where spurious vectors are identified. The relaxation process employs an augmented regularization strategy where curvature in the displacement field, consistency between neighbouring displacements, and affinity with raw measurements, are balanced. Next, invalid vectors are removed according to user defined criteria. In the next phase valid displacement vectors that were attenuated but not removed during the relaxation phase are recovered in an "*attraction*" scheme. Finally, TPS coefficients are used to represent the validated displacement field. As such, the interpolation of validated results is an inherent part of the validation process.

Preliminary results are presented for artificially generated images containing sinusoidal velocity fields. Based on these results the method holds promise as a validation strategy for application with existing PIV measurement techniques.

1.0 Introduction

The utility of Particle Image Velocimetry (PIV) for measurement of velocity fields in many flows is well established (Adrian (1991); Willert et al. (1991); Raffel et al. (1998)). It's broad applicability has created interest in overcoming deficiencies with the technique when applied to flows in which the range of velocity is wide and velocity gradients within the image field are large. A case of particular interest where these conditions are exhibited is the pool fire (O'Hern et al. (1999); Tieszen et al. (2001)). One of the major difficulties with standard cross-correlation PIV is the inherent link between the density of vectors in the measurement field and the maximum measurable displacement. If the velocity range within the field of view is large, then large areas of interest (AOIs) must be employed to capture the highest velocities, independent of the distribution of velocity in the image field. The use of large AOIs in these cases results in a sparsely populated vector displacement field with reduced spatial resolution.

Several methods exist to reduce this link between displacement and resolution. The simplest of these involves the overlap of AOIs (Raffel et al. (1998)) however, vectors produced with this method are not independent. Furthermore, it has been shown that processing images with AOI overlap where particle dispersion or image contrast is not uniform can produce a vector field that is similar to the field produced by nearest neighbour interpolation of the vector field obtained without overlap (Young et al. (2002)). More advanced strategies designed to reduce the displacement/resolution link are based on an iterative heirachical/multi-resolution scheme e.g. (Huang et al. (1993a); Huang et al. (1993b); Huang et al. (1993c); Jambumathan et al. (1995); Huang (1998); Scarano et al. (1999); Scarano et al. (2000)). A sparsely populated velocity field obtained on a first pass using large AOIs is assumed to capture the large-scale structure of the flow. Subsequent passes employ this large-scale information to measure the small-scale structure of the flow with successively smaller AOIs. These hierarchical methods have been shown to produce improved results in smoothly varying flow fields; however, these methods can falter when velocity gradients are high and/or image contrast and particle dispersion are less than optimal (Gilbert (2002)). One reason for this difficulty is that, between iterations, it can be difficult to correct for "false" or "invalid" vectors that contaminate the large-scale displacement field.

The ability of hierarchical schemes to "self correct" and attenuate the influence of erroneous vectors from previous iterations is dependent on several factors including: the range of measured displacements throughout the image field, the flow field topology, the reduction in AOI size between iterations, and details of the cross-correlation matching algorithm. In any circumstance, errors in the displacement field that are greater than ¼ of the AOI dimension used on the subsequent iteration will almost never be suitably corrected without intervention (see ¼ rule (Adrian (1991); Adrian (1997))). Two methods can be employed to mitigate the effects of "false" displacement estimates in inter-iteration results while at the same time maintaining the general hierarchical framework. These are: 1) to improve the ability of the cross-correlation algorithm to produce accurate results, and 2) to design/incorporate a validation routine to remove spurious vectors.

In previous work improvements in the accuracy of cross correlation estimates have been obtained through the development of a vector positioning scheme (Young et al. (2002)). Instead of being centred in the AOI, individual vectors are placed at locations within the AOI that correspond to dominant matching features. This produces a vector field arranged on an irregular grid. The resulting velocity field is more accurate and at the same time extends the applicability of employing AOI overlap. The algorithm is especially useful in high gradient regions and/or where contrast or particle dispersion are not uniform. It holds promise for the incorporation into hierarchical PIV analysis pursuant to the development of an appropriate validation scheme. An example result showing the effects of vector positioning is shown in figure 1 for a sinusoidal velocity profile. In this figure results are compiled on a 2-D plot for ease of analysis. The x index is normalized based on the modulus function, *mod*, and the horizontal dimension of the AOI, AOI_x. The displacement in the vertical direction, Δy, is normalized by the vertical dimension of the AOI, AOI_y. The exact distribution of displacement is indicated by the solid line. As can be seen through comparison of figure 1a) and b) the positioning strategy produces improved results where it is less difficult to visually separate valid from invalid displacement measurements. To further improve results a suitable validation scheme is needed.

In this paper, the design of a new validation framework is described. This framework allows for the incorporation of: 1) the degree of smoothness in the displacement field, 2) the similarity between local particle properties calculated from intensity information on each of the two images, and 3) the degree of certainty about the raw displacement measurements. Each of these aspects is discussed in section 2. The proposed validation strategy is discussed in section 3 and some preliminary results are shown in section 4.

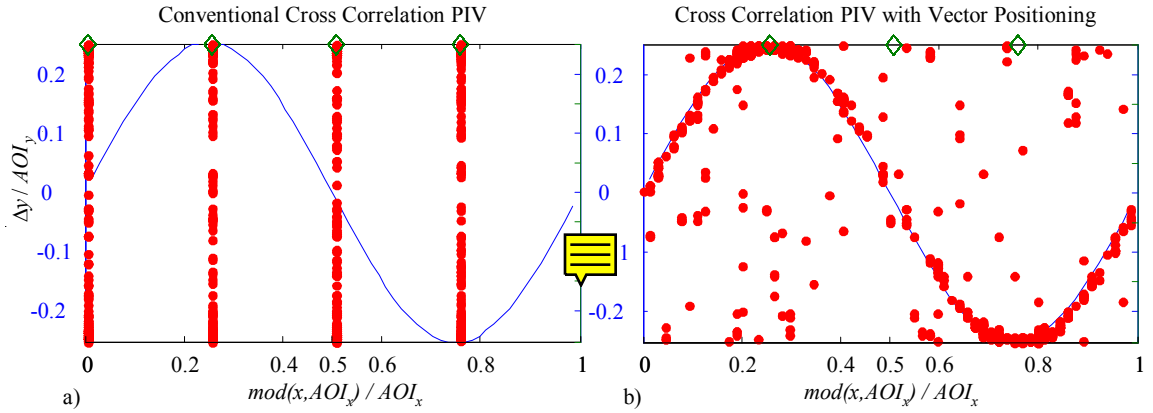


Figure 1: Example result, a sinusoidal displacement field, amplitude 16 *pix*, wavelength 64 *pix* a) standard cross correlation PIV and b) cross correlation PIV with vector positioning. AOI_x = 64 *pix*, AOI_y = 64 *pix*

2.0 Developing a Model based Validation Framework

The basis for many validation strategies is the idea that there should be a degree of topological continuity and spatial correlation between neighbouring displacements. Invoking this idea according to the relative scale of the flow in a given image and the distance between particles and/or displacement measurements, has a sound basis in the physics of fluids. Unfortunately, the motivation behind the application of PIV is usually to make determinations about the topology and the degree of correlation in the velocity field. Therefore, imposing a smoothness criteria for vector validation, although often necessary, can be somewhat tenuous and needs careful consideration. Most validation schemes are based on a first order difference or nearest neighbour similarity constraint (*e.g.* (Nogueira et al. (1997); Scarano et al. (1999))). In these, displacement vectors that do not bear resemblance with their neighbours to within a predefined threshold are identified as invalid. In smooth flows with vector fields that contain spurious invalid estimates or in cases where multiple measurements of the same steady flow field are obtained, the application of nearest neighbour methods can produce reasonable results. The application of a single threshold value implicitly assumes that within the local neighbourhood of a particle, the velocity should be approximately constant (leaving some margin for randomness and slight variation). Unfortunately, this is not always a good assumption – particularly in turbulent flows. As such, many common validation strategies are less effective when substantial velocity gradients need to be measured and/or displacement vectors do not fall on a regular grid. In these cases, the range of velocities in local regions of the field can be large making the determination of an acceptable threshold value difficult. This problem is compounded since the cross-correlation algorithm is most apt to return invalid vectors in these same regions.

A more sophisticated scheme to impose smoothness and validate results is required to enable the identification of false vectors while accepting displacement vectors in valid gradient regions. This observation points toward the application of a validation model that is more sensitive to the local topology of the measured displacement field. One might conceive of employing a model based on a conserved quantity such as mass but such models are usually not directly applicable since neither density nor the 3rd dimension of velocity are typically measured. Therefore, less exact models need to be formulated. In this work, the use of a second-order validation model based on the application of a thin plate spline (TPS) is proposed.

2.1 The TPS Smoothness Model

The TPS is a global basis function interpolant closely related to a category of interpolant functions known as "multiquadratics" (Spath (1995)). On its own, the TPS is a direct interpolant with no validation

attributes. Each measured displacement is reproduced exactly. For the collection of x and y indices on image 1, $\{\boldsymbol{x_1}, \boldsymbol{y_1}\}$ to which measured displacements $\{\boldsymbol{\Delta_x}, \boldsymbol{\Delta_y}\}$ are anchored, a set of coefficients, $\{\boldsymbol{C_x}, \boldsymbol{C_y}\}$, are determined such that the displacements may be reproduced via equation 1. Note that each collection of indices, displacements, and coefficients is represented in vector notation (*i.e.* as an Nx1 matrix). In this paper column vectors are represented with bold type. Individual elements in a vector are typically subscripted and represented with normal type.

$$\Delta x_i = \sum_{j=1} C_{xj}\, r_{i\to j}^2 \log\left(r_{i\to j}^2\right),\ \Delta y_i = \sum_{j=1} C_{yj}\, r_{i\to j}^2 \log\left(r_{i\to j}^2\right) \qquad (1)$$

In this paper the collection of position indices, $\{\boldsymbol{x_1}, \boldsymbol{y_1}\}$, are referred to as "*image 1 landmarks*". Similarly, the position indices where displacement vectors terminate are referred to as "*image 2 landmarks*". There are as many image 1 landmarks as image 2 landmarks. Based on this terminology a particle located on the first image at landmark i, (x_{1i}, y_{1i}) would be found on the second image in an image pair at the landmark with position index (x_{2i}, y_{2i}). In equation 1, $\boldsymbol{C_x}$ and $\boldsymbol{C_y}$ are the TPS coefficients defined for the x and y components of displacement respectively. The i and j indices designate a particular element in the respective column vector. $r_{i\to j}$ is the Pythagorean distance between the i^{th} and j^{th} landmarks on image 1 where displacement vectors originate. The TPS formulation minimizes the functional consisting only of the mean curvature of a free thin plate over the region R for all known displacements Δ (Franke (1982)),

$$\iint_{R^2} \left|\frac{\partial^2\Delta}{\partial x^2}\right|^2 + 2\left|\frac{\partial^2\Delta}{\partial x\partial y}\right| + \left|\frac{\partial^2\Delta}{\partial y^2}\right|^2 dx\, dy \quad . \qquad (2)$$

Outside region R^2 the TPS interpolant tends to become flat. In this TPS functional the displacements, Δ, are in the direction orthogonal to the x y plane where landmarks reside. With PIV however, displacements are measured in the same plane that landmarks are located. Therefore, the measured displacements are split into their component parts, Δx and Δy. The Δx and Δy displacements are then used separately in the TPS formulation as if they had been measured in the direction perpendicular to the x and y axis. The TPS formulation must therefore be applied twice to interpolate the measured displacement field.

The formulation in equation 1 can be extended to separate the displacement field into an affine portion and a multiquadratic warping portion (Bookstein (1989)). The affine portion accounts for global translation, shear, stretch, and rotation in the displacement field. The regular TPS portion accounts for "warpage" in the displacement field as formulated in equation 3.

$$\begin{aligned}\Delta x_i &= a_{x1} + a_{x2}x + a_{x3}y + \sum_{j=1}^{N} C_{xj}\, r_{i\to j}^2 \log\left(r_{i\to j}^2\right)\\ \Delta y_i &= a_{y1} + a_{y2}x + a_{y3}y + \sum_{j=1}^{N} C_{yj}\, r_{i\to j}^2 \log\left(r_{i\to j}^2\right)\end{aligned} \qquad (3)$$

Note that N is the number of landmark points in the image field and the coefficients a_1, a_2, a_3, are the affine coefficients. In this work the coefficients $\{\boldsymbol{C_x}, \boldsymbol{C_y}\}$ and $[a_1, a_2, a_3]^T$ are found using a least squares methodology based on the set of measured displacements.

The thin plate model is appealing since it provides a smooth displacement field that is everywhere differentiable; however, to make the model useful for the purposes of validating PIV data extra smoothness constraints are incorporated during the calculation of the coefficients, $\{\boldsymbol{C_x}, \boldsymbol{C_y}\}$. In this way the interpolant is forced to pass near but not through the measured displacements. Adding extra constraints in this fashion is known as regularization (Bertero et al. (1988); Lee (1988)). The functional to be minimized for the determination of the TPS and affine coefficients is shown in block matrix format in equation 4,

$$\arg\left|\begin{matrix}C\\A\end{matrix}\right| \min \left\{ \left|\begin{matrix}\boldsymbol{\Delta}\\ \boldsymbol{0}\end{matrix}\right| - \left|\begin{matrix}K \mid [\boldsymbol{1}\ \boldsymbol{x}\ \boldsymbol{y}]\\ I \mid [\boldsymbol{0}\ \boldsymbol{0}\ \boldsymbol{0}]\end{matrix}\right| \left|\begin{matrix}\boldsymbol{C}\\ \boldsymbol{A}\end{matrix}\right| \right\}^T \left|\begin{matrix}R & 0\\ 0 & \lambda K\end{matrix}\right| \left\{ \left|\begin{matrix}\boldsymbol{\Delta}\\ \boldsymbol{0}\end{matrix}\right| - \left|\begin{matrix}K \mid [\boldsymbol{1}\ \boldsymbol{x}\ \boldsymbol{y}]\\ I \mid [\boldsymbol{0}\ \boldsymbol{0}\ \boldsymbol{0}]\end{matrix}\right| \left|\begin{matrix}\boldsymbol{C}\\ \boldsymbol{A}\end{matrix}\right| \right\} \qquad (4)$$

where $\boldsymbol{\Delta}$ represents the vector of displacements at landmark locations $\{\boldsymbol{x}_1, \boldsymbol{y}_1\}$, $\mathbf{0}$ is a column vector of zeros, $\boldsymbol{C}$ is the vector of TPS coefficients, $\boldsymbol{A}$ is the vector of affine coefficients, $[a_1, a_2, a_3]^T$, K is the TPS kernel matrix composed of terms $r_{i\to j}^2 \log(r_{i\to j}^2)$ at index (i,j), I is the identity matrix, R is also an identity matrix, and λ is the scalar value of the regularization parameter. The form of the equation used to solve for $\boldsymbol{C}_x$ and $\boldsymbol{A}_x$ is the same as that used to solve for $\boldsymbol{C}_y$ and $\boldsymbol{A}_y$. Therefore, for the sake of compactness, the subscripts x and y are dropped from $\boldsymbol{C}$, $\boldsymbol{A}$, and $\boldsymbol{\Delta}$ in equation 4. Roughly speaking, the top most line in equation 4 represents the constraint that the coefficients $\boldsymbol{C}$ and $\boldsymbol{A}$ should produce an interpolant where measured displacements are reproduced. The bottom line represents the "low energy" constraint that tends to minimize the curvature in the interpolant (Bookstein (1989)). The magnitude of the regularization parameter, λ, reflects the balance between these two competing criteria. The main attribute of the formulation is that raw displacement vectors are not attenuated equally – the amount of attenuation is dependent on the mean curvature of the displacement field in the vicinity of a given point. An example showing the action of regularization for different values of λ is demonstrated with a simple 1-D case in figure 2. In this example image 1 landmarks, denoted with ◆ markers, are situated along the horizontal axis. Image 2 landmarks, which represent the termination points of the measured displacement vectors on image 2, are denoted with ● markers. Lines connecting ○ , ○ , ○ , ○ , are the family of TPS interpolants produced using different values of λ as specified in the legend.

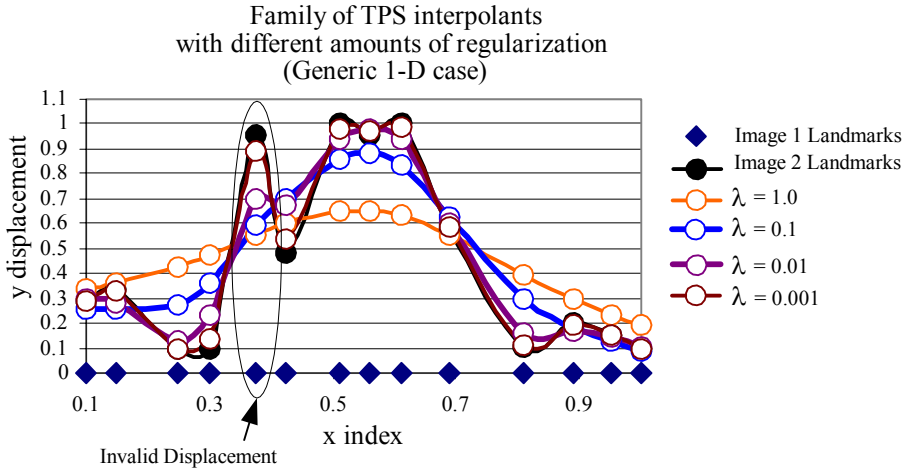


Figure 2: TPS interpolation with Regularization (y displacement only)

In this example, the raw displacements have only a vertical component and the displacement of the 5th point from the left is considered invalid. For each curve in the family of regularized interpolants in figure 2 the influence of the invalid displacement on the interpolant is attenuated. This is the desired effect. However, as indicated in the figure for a value of $\lambda = 1.0$, the action of regularization also tends to: 1) reduce the heights of hills and the depths of valleys in the interpolated field where displacements are accurate, and 2) produce an undesirable amount of attenuation for measurements on either side of the spurious displacement measurement. An augmented regularization strategy is devised to mitigate these effects. The method employed is akin to adding extra "measurement" terms to the formulation in equation 4 to effect an enhanced degree of flexibility. Specifically, the displacement at each image 1 landmark is allowed to take on several values in the least squares formulation according to a specified weighting. Each of these "extra" displacements coincides with the measured displacement at neighboring locations. The augmented version of equation 4 in matrix notation is:

$$\arg\left|\begin{matrix}\boldsymbol{C}\\ \boldsymbol{A}\end{matrix}\right| \min \left\{ \left|\begin{matrix}\boldsymbol{\Delta}'\\ \mathbf{0}\end{matrix}\right| - \left|\begin{matrix}K' \mid [\mathbf{1}\ \boldsymbol{x}\ \boldsymbol{y}]\\ I \mid [\mathbf{0}\ \mathbf{0}\ \mathbf{0}]\end{matrix}\right| \left|\begin{matrix}\boldsymbol{C}\\ \boldsymbol{A}\end{matrix}\right| \right\}^T \left|\begin{matrix}R' & 0\\ 0 & \lambda K\end{matrix}\right| \left\{ \left|\begin{matrix}\boldsymbol{\Delta}'\\ \mathbf{0}\end{matrix}\right| - \left|\begin{matrix}K' \mid [\mathbf{1}\ \boldsymbol{x}\ \boldsymbol{y}]\\ I \mid [\mathbf{0}\ \mathbf{0}\ \mathbf{0}]\end{matrix}\right| \left|\begin{matrix}\boldsymbol{C}\\ \boldsymbol{A}\end{matrix}\right| \right\} \qquad (5)$$

where K′ is an Nx1 tiled version of the TPS kernel matrix K, R′ is a diagonal matrix where elements along the main diagonal are the elements of a weight vector $\boldsymbol{M}'$, and $\boldsymbol{\Delta}'$ is the augmented displacement vector. $\boldsymbol{M}'$ and $\boldsymbol{\Delta}'$ both result from a lexicographic reorganization of a symmetric matrix (i.e. piling each of the columns in the matrix on top of one another). Equation 6 demonstrates the formulation of K, $\boldsymbol{\Delta}'$ and $\boldsymbol{M}'$.

$$K' = \begin{vmatrix} K \\ K \\ K \\ . \\ . \\ . \end{vmatrix}, \quad \Delta' = lex \begin{vmatrix} \Delta_1^T \\ \Delta_2^T \\ \Delta_3^T \\ . \\ . \end{vmatrix}^T = \begin{vmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \\ . \\ . \end{vmatrix}, \quad M' = lex \begin{vmatrix} \boldsymbol{M}_1^T \\ \boldsymbol{M}_2^T \\ \boldsymbol{M}_3^T \\ . \\ . \end{vmatrix}^T = \begin{vmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \\ \boldsymbol{M}_3 \\ . \\ . \end{vmatrix} \tag{6}$$

In equation 5, $\boldsymbol{\Delta}'$ enables the displacement at each image 1 landmark to take on the measured displacement at every other landmark in the field according to the weighting specified by the vector $\boldsymbol{M}'$. In equation 6 each $\boldsymbol{\Delta_i}$ is an Nx1 column vector where all elements take on the value of the i^{th} element in the raw measurement vector $\boldsymbol{\Delta}$. Each $\boldsymbol{M_i}$ is also an Nx1 column vector. However in this case the j^{th} element in $\boldsymbol{M_i}$ represents a weight coefficient between the j^{th} landmark on image 1 and the measured displacement at the i^{th} landmark on the same image.

The details of implementing this augmented regularization formulation for the purposes of validation are explained in section 3. However, before proceeding to this discussion it is important to note that $\boldsymbol{M}'$ and R′ can be designed to reflect 1) the similarity between particle local properties calculated from local intensity information on each of the two images, and 2) the degree of certainty about the raw displacement measurement before validation.

3.0 A TPS model based Validation strategy

The proposed validation scheme is divided into 5 parts. In part 1, after the measurement of the raw displacement field, a number of control parameters are initialized and a regularized displacement field is calculated using equation 4. This regularized displacement field is used to initialize an iterative "relaxation" stage designed to identify spurious "invalid" vectors through repeated use of the augmented TPS formulation in equation 6. In part 3 the invalid vectors are removed. In part 4, termed the "*attraction*" phase, valid displacement vectors that were attenuated but not removed during the relaxation phase are redefined. Part 5 is the final validation step where the TPS coefficients calculated at the end of part 4 are used to represent the validated displacement field. The algorithm layout is depicted in figure 3. Details pertaining to each of the 5 parts are discussed in the subsequent sections.

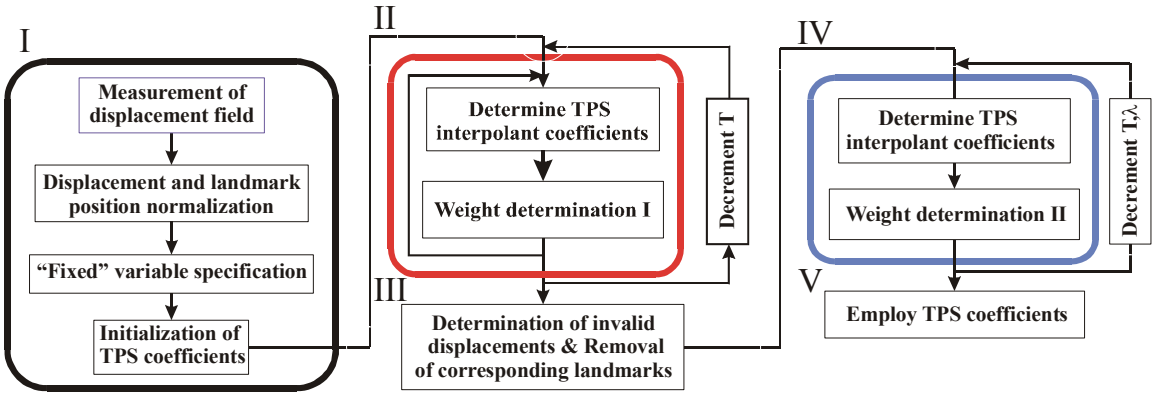


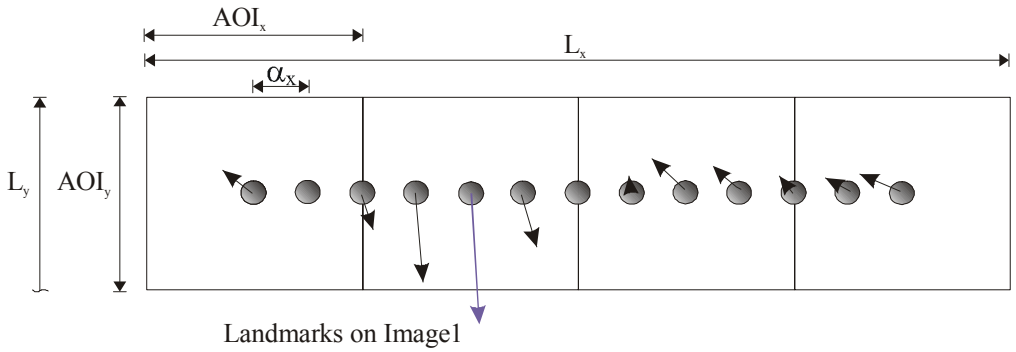
Figure 3: TPS validation algorithm: I initialization, II relaxation, III removal, IV attraction, V application

3.1 Program Initialization

The first stage in the validation algorithm is an initialization stage where: 1) measured displacements and landmark position indices are normalized, 2) several "fixed variable" constants are specified based on

available knowledge about the flow field, and 3) a regularized version of the raw displacement field is calculated.

Since the TPS formulation is sensitive to the dimensions used during the calculation of $r_{i \to j}$ in the TPS kernel, K, landmark positions and measured displacements are normalized to maintain a degree of consistency between different imaging scenarios. The indices of Image 1 landmarks are normalized between 0 and 1 based on the maximum image dimension. Measured displacements are normalized based on the maximum image dimension, the AOI dimension, and the amount of overlap applied during the cross-correlation sweep through the image. Image 2 landmarks are calculated by adding the normalized displacements to the normalized image 1 landmarks. The equations and nomenclature for this normalization procedure are shown in figure 4.



Landmarks on Image1

$$x_1 = \frac{x_1}{\max(L_x, L_y)}, \quad y_1 = \frac{y_1}{\max(L_x, L_y)}$$

Displacements from Image1 to Image2

$$\Delta_x = \frac{\Delta_x \, \alpha_x}{0.25 \, AOI_x \max(L_x, L_y)}, \quad \Delta_y = \frac{\Delta_y \, \alpha_y}{0.25 \, AOI_y \max(L_x, L_y)}$$

Landmarks on Image2

$$x_2 = x_1 + \Delta_x, \quad y_2 = y_1 + \Delta_y$$

Nominal distance between points on Image1

$$dx_n = \frac{AOI_x}{L_x \, \sigma_x}, \quad dy_n = \frac{AOI_y}{L_y \, \sigma_y}$$

Maximum Normalized Displacement

$$\Delta x_n = \frac{\alpha_x}{L_x}, \quad \Delta y_n = \frac{\alpha_y}{L_y}$$

Figure 4: Normalization of landmark indices and displacement measurements. L_x and L_y are the image dimensions in the horizontal and vertical direction, AOI_x and AOI_y are the AOI dimensions in the horizontal and vertical direction, α_x and α_y (not shown) are the distances between neighbouring AOIs in the horizontal and vertical direction, σ_x and σ_y are user defined "geometric compression" constants, and the subscript n stands for the designation "nominal".

After normalization of landmark indices and measured displacements several "fixed variable" constants are specified. Effort has been made to reduce the number of controlling fixed variable constants to make implementation of the validation algorithm easier and more robust. Two user-defined quantities that play major roles in the validation algorithm are the regularization parameter, λ, and the region of support.

The regularization parameter, λ, controls the proportion through which the TPS "energy" term governs the calculation of the TPS coefficients $\boldsymbol{C}$ and $\boldsymbol{A}$. In the program initialization phase, λ is used to calculate the simple regularized flow field according to equation 4. This regularized flow field becomes the starting point from which invalid displacements are identified. λ remains set at its initial value during parts II and III of the algorithm.

The region of support is used to define the region of influence characterized by a mask matrix Q used in parts II, III, and IV of the algorithm. This mask matrix is such that each element, q_{ij}, represents the distance from the j^{th} landmark on image 1 to the i^{th} landmark on the same image. Values of q_{ij} greater than a pre-specified region of support are set to zero; the rest of the elements are set to 1. The Q matrix is used to set the influence of raw displacements for landmark points outside of a specified region of support to zero. In this algorithm a rectangular region of support is employed. The dimensions of this region may be chosen based on prior information about the flow field. (More advanced adaptations of the algorithm may incorporate an elliptical region of support where the orientation and extent of the principal axes could be made position dependant if prior information regarding the correlation structure of the flow field is available.) Within the region of support the degree of consistency between valid displacement vectors can also be specified via "geometric compression" variables σ_x and σ_y. Setting $\sigma_x = 1$ and $\sigma_y = 2$, for example, would indicate that displacements 1 unit apart in the x direction should have the same degree of similarity as displacements 2 units apart in the y direction. The geometric compression variables are used in the calculation of the normalization quantities, dx_n, and dy_n, that describe the nominal distances in each component direction between image 1 landmarks. The formulation for dx_n, and dy_n is shown in figure 4. Also shown in figure 4 is the formulation for the maximum nominal displacement in each component direction, Δx_n and Δy_n. Quantities, dx_n, dy_n, Δx_n and Δy_n, are used during the calculation of $\boldsymbol{M'}$ weight terms in steps II, III, and IV of the algorithm.

3.2 Relaxation

After the calculation of the regularized flow field in part I, the formulation is augmented with "extra" displacement measurements as discussed in Section 2.1. The core issue here is the calculation of appropriate weight terms in $\mathbf{M'}$ to accompany the use of these extra displacements.

The incorporation of "extra" displacement information is based on two observations that coincide with the use of regularization alone: 1) invalid vectors are attenuated more than their neighbours, and 2) invalid displacement vectors are attenuated to become more similar to their neighbours. Based on the family of regularization curves in figure 2, the weighting formulation described by the vector $\boldsymbol{M'}$ is designed to 1) further decrease the influence of invalid vectors, 2) retain a close reproduction of overall topology of the displacement field in smoothly varying regions. The methodology to determine $\boldsymbol{M'}$ begins with forming a matrix where each element represents the distance, $d_{i \to j}$, from the regularized position of the j^{th} landmark from image 2 to the i^{th} landmark on image 2. For purposes of discussion the regularized position indices of image 2 landmarks are denoted as "*iterate landmarks*" since the position of these landmarks changes between iterations in the relaxation phase of the algorithm. Elements along the diagonal of this $d_{i \to j}$, matrix (i.e. $d_{i \to i}$) correspond to the distance between a raw (original) image 2 landmark and its corresponding iterate landmark location. This is shown graphically in figure 5.

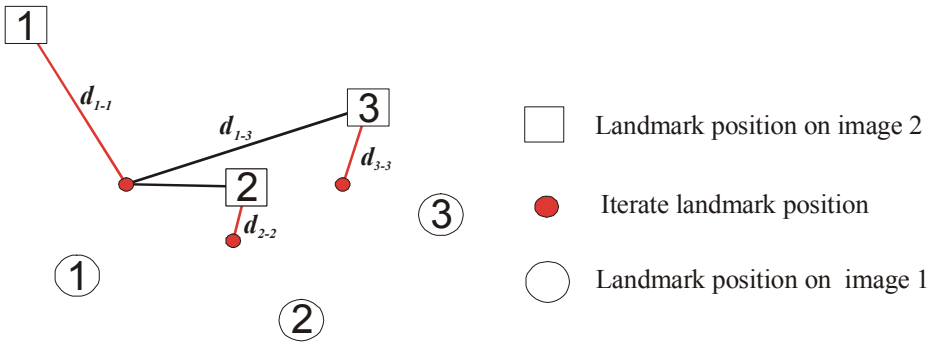


Figure 5: Nomenclature for calculation of $\boldsymbol{M'}$

The $d_{j \to i}$ elements along the main diagonal are normalized by the nominal distances Δx_n and Δy_n according to equation 7 where $dx_{j \to i}$ is the component of $d_{j \to i}$ in the horizontal direction and $dy_{j \to i}$ is the component of $d_{j \to i}$ in the vertical direction. $d_{j \to i}$ elements that do not fall on the diagonal are normalized via dx_n and dy_n in a similar fashion.

$$d_{j\to i} = \begin{cases} \left(\left(\frac{dx_{j\to i}}{dx_n} \right)^2 + \left(\frac{dy_{j\to i}}{dy_n} \right)^2 \right)^{\frac{1}{2}} & \forall\ j \neq i \\ \left(\left(\frac{dx_{j\to i}}{\Delta x_n} \right)^2 + \left(\frac{dy_{j\to i}}{\Delta y_n} \right)^2 \right)^{\frac{1}{2}} & \forall\ j = i \end{cases} \tag{7}$$

Recall that i represents the row index and j represents the column index in the matrix containing $d_{j\to i}$ elements. This normalization procedure accounts for the effects of AOI overlap used in obtaining the raw measurements. Once normalized, the $d_{j\to i}$ matrix is reformulated into a new matrix, D, where individual elements are defined according to equation 8.

$$D_{j\to i} = q_{ij} \exp\left(\frac{-\sqrt{d_{j\to i}^2 + d_{i\to i}^2 + p_{j\to i}^2}}{T} \right) \tag{8}$$

This formulation provides an intuitive inverse relationship in which small values of $\sqrt{d_{j\to i}^2 + d_{i\to i}^2 + p_{j\to i}^2}$ return large values of $D_{j\to i}$. The $p_{i\to j}$ term in equation 8 is added to show where a property correspondence metric can be incorporated into the weighting formulation. An exponential function is used in lieu of a simple inverse function so that the degree of inverse dependence can be independently controlled. A low value of T provides a much higher weighting to small values of $\sqrt{d_{j\to i}^2 + d_{i\to i}^2 + p_{j\to i}^2}$. If T is high the difference between values of $D_{j\to i}$ is reduced. Once the D matrix is calculated, elements are normalized by the sum of elements in the same column.

$$D_{j\to i} = \frac{D_{j\to i}}{\sum_{n=1}^{N} D_{j\to n}} \tag{9}$$

The transpose of D is then lexicographically decomposed into the $\boldsymbol{M'}$ vector, (*i.e.* $\boldsymbol{M'} = \text{lex}[D^T]$). Once $\boldsymbol{M'}$ is determined, elements are placed along the main diagonal of R′ in equation 5 and the TPS coefficients for both the x and y components of displacement are determined. New positions for iterate landmarks are then calculated and the relaxation procedure is iterated. In theory, for a set value of T, iterate landmarks will continue to move until convergence, after which the value of T may be reduced. However, in practice convergence need not be achieved for each value of T. Between 2 and 4 iterations at each value of T has shown to provide acceptable results in an expedient fashion. This alternating update process is very similar to that of deterministic annealing described in Gold et al. (1998) and Stellmacher et al. (2000).

In the present algorithm the initial value of T is set (*i.e*, before the first iteration) to a value representing the 90$^{\text{th}}$ percentile of the values of $\sqrt{d_{j\to i}^2 + d_{i\to i}^2 + p_{j\to i}^2}\ \forall\ j = i$. The final value of T used during relaxation is set to the $(0.75/N)^{\text{th}}$ percentile of the values of $\sqrt{d_{j\to i}^2 + d_{i\to i}^2 + p_{j\to i}^2}\ \ \forall j,i$. Recall that N is the number of points in the image field and the $D_{j\to i}$ matrix has NxN elements. Between these two endpoints the value of T is set to vary linearly with iteration number. Setting the endpoint values of T in this manner reduces the size of the effective region of support around every iterate landmark as the relaxation process progresses. It also makes the value of T dependent on both the value of λ and the topology of the measured flow field.

The relaxation phase of the validation algorithm produces a displacement field where invalid raw estimates of displacement are replaced by displacement vectors that are functions of the displacements at neighbouring locations according to TPS smoothness. The method has been shown to work well even in regions of high gradient. Figure 6 shows the TPS interpolant produced after each relaxation iteration for the example case presented in figure 2 (λ = 0.001). Note that the invalid displacement is corrected and valid displacements are smoothed slightly during this phase of the algorithm.

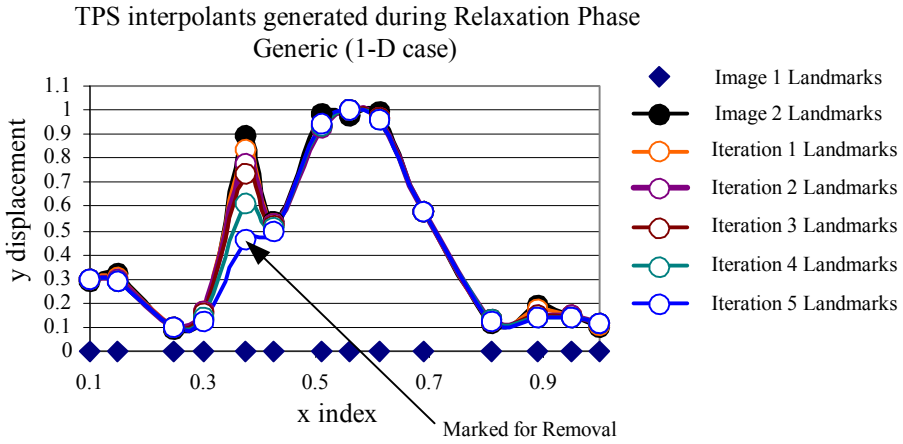


Figure 6: TPS interpolants produced during Relaxation. Invalid point marked for removal.

3.3 Point Removal

Since there are as many TPS coefficients as there are measured vectors, it is necessary to remove any landmarks that correspond to invalid vectors. This is done by computing the $D_{j\to i}$ matrix (equation 8) without normalizing by the sum of the elements in each column (equation 9). Then the ratio of the diagonal element in each column to the maximum element in the same column is computed. If this ratio is less than $exp(-\beta)$ (where β is typically set between 0.25 and 1.5 depending on the strictness of the validation requirement), the corresponding landmarks on image 1 and 2 are tagged for removal. (*i.e.* displacement vectors that correspond to tagged landmarks are deemed invalid. This procedure is repeated for each column in the $D_{j\to i}$ matrix. In figure 6 the 5th point from the left is determined to be invalid. Once invalid landmark points are identified, their influence in the TPS formulation is removed and the TPS displacement coefficients are recalculated using equation 5 based on the value of T set at the end of the relaxation phase.

At this point the validation process may be terminated and the TPS coefficients may be used to determine the displacement at any point in a field in which displacements are both validated and smoothed. In some cases, results from a refinement step in the hierarchical PIV process can sometimes be improved if some of the high frequency information is removed from the field produced in the previous step. If the discrepancies between actual and smoothed displacements are less than ¼ of the dimension of the AOI used in the next step of the PIV measurement scheme, actual displacements can still be recovered. The challenge is to provide a mechanism to actively control the smoothness in the validated displacement field. Unfortunately, the value of λ which controls the degree of smoothness in the validated displacement field, also controls point removal. Tailoring λ to achieve optimal results for both operations while at the same time making the validation algorithm relatively independent of the flow field topology is a difficult task. Instead, the value of λ is reduced in an iteration scheme that commences after point removal. This part of the validation process, labeled the attraction phase, is terminated when iterate landmarks are *all* within a certain distance of their image 2 locations.

3.4 The Attraction Phase

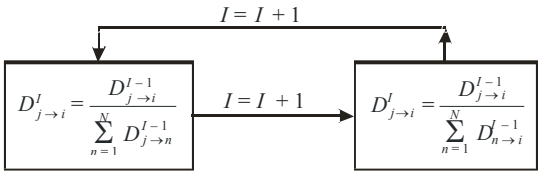
The attraction phase is designed to take the iterate landmarks towards image 2 landmarks in a stepwise fashion during which time the process may be halted based on a chosen criterion. A simple method to achieve this would be to move each iterate landmark along a line connecting it to it's corresponding landmark point on image 2. However, the result is often not as smooth as expected. Instead, a process that mirrors that of the relaxation phase is used. First, iterate landmark positions are used to calculate an $\boldsymbol{M'}$ vector; then the formulation in equation 5 is used to determine new TPS coefficients. These coefficients are then used to recalculate new positions for the iterate landmarks and the process is repeated. The main differences between attraction and relaxation are: 1) the formulation for calculating the weights is slightly different, and 2) the values of λ and T are reduced toward 0 between iterations. For validation between intermediate steps of a multi-pass cross correlation analysis strategy a logical criterion is to stop the

attraction process when the distance between each iterate landmark and its corresponding image 2 landmark is less than 0.25 of the AOI dimension used in the next hierarchical step in the PIV measurement scheme (*i.e.* using the ¼ rule).

As before, the methodology to determine $\boldsymbol{M'}$ begins with formulation of a matrix where each element $d_{j\to i}$ represents the distance from the position of the j^{th} iterate landmark to the ith landmark on image 2. And, as before, these distances are normalized via equation 7 and reformulated into a D matrix. However the equation for each of the elements in the D matrix simplifies to:

$$D_{j\to i} = q_{ij} \exp\left(\frac{-\sqrt{d_{j\to i}^2 + p_{j\to i}^2}}{T}\right) . \tag{10}$$

Once compiled, the $D_{j\to i}$ matrix is normalized; however, compared to the relaxation phase, the normalization procedure is more involved. First each element is divided by the sum of the elements in the column in which it is situated. Elements in the resulting matrix are then divided by the sum of the elements in the row in which the element sits. The process is then repeated in an iterative fashion for a preset number of iterations or until convergence is reached. The algorithm is shown in equation 11 where I indicates the iteration number.



(11)

It has been shown by Sinkhorn (1964) that this algorithm will produce a stochastic matrix where the sum of each row and each column adds to 1. A similar algorithm has been used by Chui et al. (2000) in an iterative point correspondence strategy. As T becomes smaller the matrix begins to resemble a permutation matrix where the existence of a 1 indicates an exclusive match between the iterate landmark identified by the column index j and the image 2 landmark identified by the row index i. In the present algorithm Sinkhorn normalization is employed to obtain a 1 to 1 correspondence between appropriate iterate and image 2 landmarks. After normalization the resulting matrix is lexicographically decomposed to form $\boldsymbol{M'}$, $\boldsymbol{M'} = \text{lex}[D^T]$. The TPS coefficients are then recalculated, the $d_{j\to i}$ matrix is formulated, and the process is iterated. Figure 7 shows the resulting displacement field after successive attraction iterations. In this case the validated raw displacements are reproduced exactly on the final iteration.

3.5 Displacement interpolation

The final step in the validation process is the use of the calculated TPS coefficients to estimate the displacement at any position in the image field. Note that interpolation of the validated displacement field, which is necessary in many hierarchical PIV strategies, is an implicit consequence of the TPS validation formulation.

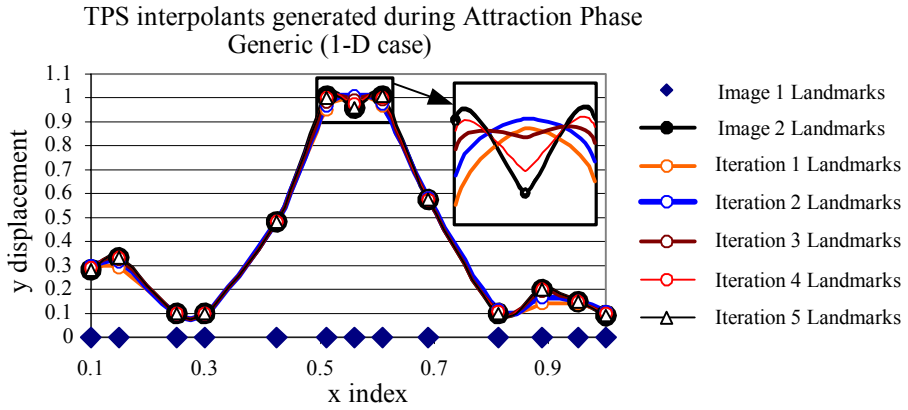


Figure 7: TPS interpolants produced during "Attraction": Invalid point removed

4.0 Some exemplary results

To demonstrate the effectiveness of the proposed validation strategy results from the analysis of synthetic images representing flows with 2 different sinusoidal velocity fields are illustrated below. Figure 8 corresponds to the validation of results in figure 1 (wavelength 64pix, amplitude 16pix). Figure 9 corresponds to the validation of results for a sinusoidal field of wavelength 128pix, amplitude 16pix. Since there is no displacement in the vertical direction, information can be compressed onto a 2-D plot. In each case the attraction phase is halted before iterate landmarks converge at image 2 landmarks. This accounts for the slight discrepancy between valid raw displacements and the displacements returned after validation. These intermediate results show promise for adaptation of the algorithm within hierarchical PIV algorithms when large gradients in velocity exist. For future adaptations of the algorithm, efforts will focus on: 1) the use of appropriate metrics for the inclusion of point properties and displacement uncertainties, 2) an efficient iterative methodology for the computation of the least squares solution (equation 5), 3) robust methodologies to determine appropriate values for the regularization parameter λ, the vector removal criterion β, and the parameter T, and 4) alternate strategies for calculating weight terms (*i.e.* $\boldsymbol{M'}$) in the relaxation phase.

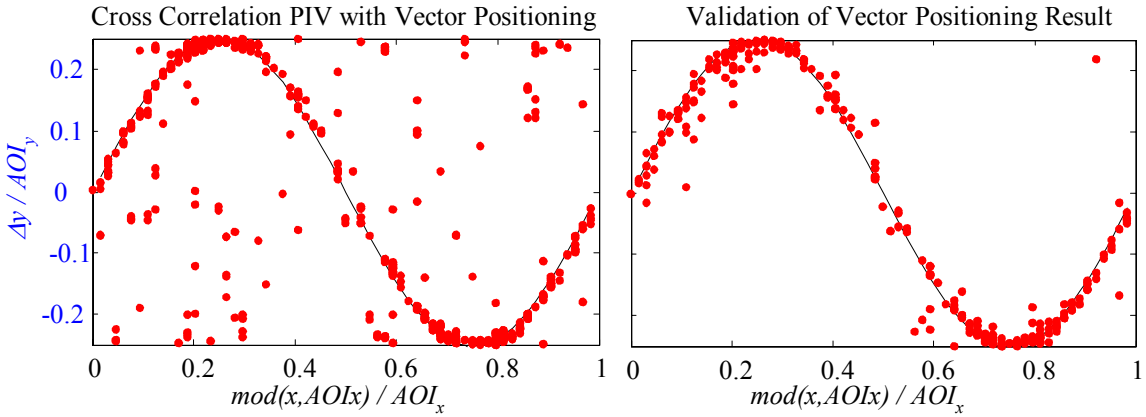


Figure 8: Validation of PIV results in Figure 2: $\lambda = 0.01$, $\beta = 0.25$

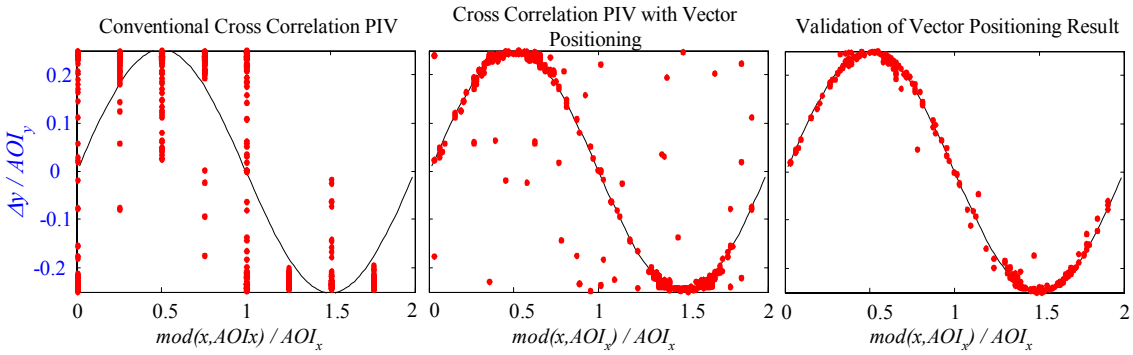


Figure 9: Validation of sinusoidal displacement field (amplitude 16 *pix*, wavelength 128 *pix*, AOI_x = 64 *pix* AOI_y = 64 *pix),* $\lambda = 0.001$, $\beta = 0.5$.

5.0 Summary

The framework for a new "model" based PIV validation strategy has been outlined. It is applicable to PIV measurements in flows with high velocity gradients and/or where displacement vectors are not positioned on a regular grid. The methodology has been designed to incorporate: 1) the degree of smoothness in the displacement field, 2) the similarity between particle properties calculated from local intensity information on each of the two images, and 3) the degree of certainty about the raw displacement measurement before validation. The utility of the methodology has been demonstrated for two different velocity profiles where valid displacement range between -¼ and +¼ of the of the AOI dimension. These preliminary results were obtained using only the smoothness constraint. A validation algorithm based on the proposed framework shows promise for inclusion between iterations in hierarchical PIV strategies that use cross correlation matching. Future efforts will be directed toward refining the implementation of the algorithm for development of a universal validation strategy applicable to flows with large velocity gradients where prior information may or may not be known about the velocity field.

6.0 References

Adrian, R. J. (1997). "Dynamic Ranges of Velocity and Spatial Resolution of Particle Image Velocimetry." Measurement Science and Technology 8: 1393-1398.

Adrian, R. J. (1991). "Particle-Imaging Techniques of Experimental Fluid Mechanics." Annual Review of Fluid Mechanics 23: 261-304.

Bertero, M., T. A. Poggio and V. Torre (1988). "Ill-Posed Problems in Early Vision." Proceedings of the IEEE 76(8): 869-889.

Bookstein, F. L. (1989). "Principal Warps: Thin Plate Splines and the Decomposition of Deformations." IEEE Transactions of Pattern Analysis and Machine Intelligence 11(6): 567-585.

Chui, H. and A. Rangarajan (2000). A New Algorithm for Non-Rigid Point Matching. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Hilton Head Island South Carolina.

Franke, R. (1982). "Scattered Data Interpolation: Tests of Some Methods." Mathematics of Computation 38(157): 181-200.

Gilbert, R. A. D. (2002). Evaluation of FFT Based Cross-Correlation Algorithms for Particle Image Velocimetry. MASc Thesis: Department of Mechanical Engineering, Waterloo, Canada, University of Waterloo.

Gold, S., A. Rangarajan, S.-P. Lu, S. Pappu and E. Mjolsness (1998). "New Algorithms of 2D and 3D point Matching: Pose Estimation and Correspondence." Pattern Recognition 31(8): 1019-1031.

Huang, H. (1998). “An Extension of Digital PIV-Processing to Double-Exposed Images.” Experiments in Fluids 24: 364-372.

Huang, H. T. and H. E. Fiedler (1993a). “Deformed Particle Image Pattern Matching in Particle Image Velocimetry.” Applied Scientific Research 51: 179-183.

Huang, H. T., H. E. Fiedler and J. J. Wang (1993b). “Limitation and Improvement of PIV, Part I: Limitation of Conventional Techniques due to Deformation of Particle Image Patterns.” Experiments in Fluids 15: 168-174.

Huang, H. T., H. E. Fiedler and J. J. Wang (1993c). “Limitation and Improvement of PIV, Part II: Particle Image Distortion, A Novel Technique.” Experiments in Fluids 15: 168-174.

Jambumathan, K., X. Y. Ju, B. N. Dobbins and S. Ashforth-Frost (1995). “An Improved Cross-Correlation Technique for Particle Image Velocimetry.” Measurement Science and Technology 6: 507-514.

Lee, D. (1988). “Some Computational Aspects of Low-Level Computer Vision.” Proceedings of the IEEE 76(8): 890-898.

Nogueira, J., A. Lecuona and P. A. Rodriguez (1997). “Data Validation, False Vectors Correction and Derived Magnitudes Calculation on PIV Data.” Measurement Science and Technology 8: 1493-1501.

O'Hern, T. J., S. R. Tieszen, R. W. Schefer, A. L. Gerhart, C. N. Young and E. J. Weckman (1999). “Simultaneous Measurment of Velocity and Concentration Fields Using a High Speed Motion Picture PIV/PLIF System in a Buoyant Helium Plume.” .

Raffel, M., C. Willert and J. Kompenhans (1998). Particle Image Velocimetry. Berlin, Springer.

Scarano, F. and M. L. Riethmuller (1999). “Iterative Multigrid Approach in PIV Image Processing with Discrete Window Offset.” Experiments in Fluids 26: 513-523.

Scarano, F. and M. L. Riethmuller (2000). “Advances in iterative multigrid PIV mage processing.” Experiments in Fluids [Suppl.]: S51-S60.

Sinkhorn, R. (1964). “A Relationship between Arbitrary Positive Matricies and Doubly Stochastic Matricies.” Annals of Mathematical Statistics 35: 876-879.

Spath, H. (1995). Two Dimensional Spline Interpolation Algorithms. Massachusetts, A.K. Peters Pub..

Stellmacher, M. and K. Obernayer (2000). “A New Particle Tracking Algorithm based on Deterministic Annealing and Alternative Distance Measures.” Experiments in Fluids 28: 506-518.

Tieszen, S. R., T. J. O'Hern, R. W. Schefer, E. J. Weckman and T. K. Blanchat (2001). “Experimental Study of the Flow Field In and Around a One Meter Diameter Methane Fire", paper # CF01-013, Combustion and Flame (in review): 20 pp.

Willert, C. E. and M. Gharib (1991). “Digital Particle Image Velocimetry.” Experiments in Fluids 10: 181-193.

Young, C. N., R. Gilbert, D. A. Johnson and E. J. Weckman (2002). Vector Positioning to Reduce Positional Bias and Improve PIV Results (FEDSM2002-31171). Proceedings of the ASME, FEDSM'02, Flow Visualization and Laser Anemometry Symposium, 2002 ASME Fluids Engineering Division, Summer Meeting, July 14-18, Montreal, Quebec, Canada.