Experiments in Fluids (2006) 40: 764–775
DOI 10.1007/s00348-006-0113-1

RESEARCH ARTICLE

**Shubhra K. Misra · M. Thomas · C. Kambhamettu
J. T. Kirby · F. Veron · M. Brocchini**

# Estimation of complex air–water interfaces from particle image velocimetry images

Received: 13 April 2005 / Revised: 22 September 2005 / Accepted: 30 September 2005 / Published online: 10 February 2006


**Abstract** This paper describes a method for the estimation of the instantaneous air–water interface directly from particle image velocimetry (PIV) images of a laboratory generated air entraining turbulent hydraulic jump. Image processing methods such as texture segmentation based on gray level co-occurrence matrices are used to obtain a first approximation for the discrete location of the free surface. Active contours based on energy minimization principles are then implemented to get a more accurate estimate of the calculated interface and draw it closer to the real surface. Results are presented for two sets of images with varying degrees of image information and surface deformation. Comparisons with visually-interpreted surfaces show good agreement. In the absence of in-situ measurements, several verification tests based on physical reasoning show that the free surface is calculated to acceptable levels of accuracy. Aside from a single image used to tune the set of parameters, the algorithm is completely automated to process an ensemble of images representative of typical PIV applications. The method is computationally efficient and can be used to track fluid-interfaces undergoing non-rigid deformations.

S. K. Misra (✉) · J. T. Kirby
Center for Applied Coastal Research, University of Delaware, Newark, DE 19716, USA
E-mail: shubhra.misra@gmail.com

M. Thomas · C. Kambhamettu
Video/Image Modeling and Synthesis Lab., Department of Computer and Information Sciences, University of Delaware, Newark, DE, USA

F. Veron
Graduate College of Marine Studies, University of Delaware, Newark, DE, USA

M. Brocchini
D.I.A.M, Universita' di Genova, Via Montallegro 1, 16145 Genova, Italy

## 1 Introduction

Particle image velocimetry (PIV) has become an established non-intrusive measurement technique to measure the kinematics of turbulent fluid flow in controlled laboratory experiments. The flow is seeded with suitable tracer particles, illuminated by a planar laser sheet, and time-lapsed images are recorded. The displacement of the particle images is measured in the plane of the image, and used to determine the displacement of the particles in the flow (Raffel et al. 2001). Frequently, these studies focus on flows where the air–water interface is free (e.g., gravity wave breaking; Qiao and Duncan 2001) or forced (e.g., wind shear over gravity waves; Donelan et al. 2004). It is often desirable to obtain detailed instantaneous flow velocities in either or both phases near the interface. This necessitates an accurate (to at least the measurement resolution of the velocities) estimation of the interface. At times, the spatio-temporal characteristics of the interface itself are of interest. Typically, the scalar interface is concurrently visualized by a technique called laser induced fluorescence (LIF) in which fluorescent dye is added to one phase and excited to a particular wavelength by the laser. LIF, in addition to its use as a technique to visualize the flow, is also used to measure the scalar concentrations (Dewey 1976). In the context of determining the scalar interface, the LIF images are transformed into binary images by a threshold-detection method and the sharp gradient in the binary image yields the interface (Westerweel et al. 2002). There are, however, various associated complications with evaluating scalar interfaces of turbulent flows. For example, the treatment of false interfaces resulting from moderate to high intensity turbulence fluctuations needs specific thresholding methods applied to the squared gradient and the Laplacian of pixel intensities (Prasad and Sreenivasan 1989).

A simultaneous PIV and LIF experiment therefore requires two separate imaging systems: one camera, with a suitable filter, which records only the particular

wavelength of light scattered by the seeding particles to measure the flow velocities, and the other, recording the light emitted by the dye used to visualize the interface. Recently, in addition to obtaining flow velocities, there have been attempts to estimate the interfaces in fluid–fluid flows directly from the PIV images. Due to the voluminous amount of images generated in a typical PIV experiment, the manual calculation of the interface remains a daunting (and owing to the subjectivity of human perception, an ambiguous) task. Hassan et al. (1996) and Okamoto et al. (1996) identified the interface by tracking illuminated floating particles which, due to their own degrees of freedom, cause frequent interruptions in the calculated surface and might even interfere with the flow. The high density of particles required to obtain a clear visualization of the surface contaminated the surface and changed the surface tension at the interface. Peirson (1997) used a multiple pulsed single frame PIV technique to study the viscous sub-layer at a wavy air–water interface. Two separate cameras were used to simultaneously measure the wave phase and the particle velocities. However, this method cannot be used to track the evolution of a time-dependent surface. The author also points out that reliable measurements were not possible in regions of small surface undulations. The presence of clear particle reflections which could be correlated to sub-surface particles was an essential requirement in the acceptability of a sub-surface image. No comparisons of the interface location with independent ground-truth measurements were reported. Law et al. (1999) used a fluorescent dye and a gradient based edge detection routine along with optimally adjusted multiple viewing mirrors to estimate the shear-free surface for a confined submerged jet in a tank. The calculated root mean square interface velocities compared reasonably well with single point wave gage measurements. Intensity or luminance based edge detection methods fail when spurious particle reflections are present in the PIV image. This problem is exacerbated in the presence of surface irregularities and cross-plane variations. To overcome similar difficulties, Lin and Perlin (1998) describe a customized water channel and optics configuration set-up based on the Brewster angle for the air–water interface. They point out that their set-up could lead to image distortion which has to be corrected for. The method is labor intensive as it involves extensive trial and error arrangements. No comparisons with independent measurements were made. Some researchers have also used a posteriori methods to determine the interface after the velocities have been calculated. Tsuei and Savas (2000) have proposed a first-order accurate method to calculate interfaces in fluid–fluid flows. Their method utilizes the velocity information embedded in a pair of PIV images to extend the image data fields across the interface. This is done by locally translating and reflecting across the interface and reversing the order of images. However, when dealing with moving or compliant interfaces, the velocities at material points on the interfaces have to be supplied as an input to the algorithm. Hirsa et al. (2001) have used the assumption of total internal reflection of the seeding particles about the true interface, which acts like a mirror, in locating the interface. The method makes use of the similarity between the true and reflected regions by correlating the patterns, and it is noted that the orientation of the light source and the camera can cause distortion of the reflected image as well as a difference between the brightness of the particle and its surface reflection. Careful and empirical adjustments of both the laser light sheet and the camera viewing angle are required to ensure clear reflections of particles for pattern matching. The correlation estimates were calculated at every pixel in the vertical direction by overlapping the windows. For the case of gravity-capillary waves, independent surface elevation measurements from a micrometer stage needle showed maximum average wave amplitude errors of approximately 5 pixels. Dabiri and Gharib (1997) used a no-shear boundary condition on the calculated velocity to determine the free surface in their experiments with spilling breaking waves. It is obvious that such methods imposing kinematic or dynamic constraints on the interface cannot be applied universally.

There is thus a clear need for a robust, objective and automated method which would be able to calculate the interface solely based on the information available in a typical PIV image. It is understood, as will become clear further in the paper, that the level of accuracy achievable by the proposed method will not be comparable to that offered by LIF. The current methodology is therefore not implied as a replacement for LIF in determining the scalar interface. Instead, given a typical ensemble of PIV images, the aim is to calculate the instantaneous interface efficiently and with a reasonable degree of accuracy. A consequential benefit is the exclusion of possible spurious particle reflections arising from a standard optical set-up (see Fig. 2) which contribute a finite but false correlation in the velocity estimation process. This would, further, obviate the extensive preliminary adjustments required for the imaging equipment as also the need to impose physical constraints on the flow kinematics when estimating the interface.

Texture classification and segmentation methods such as those based on gray level co-occurrence matrices (GLCM) (Haralick et al. 1973), cross-diagonal texture matrix methods (Al-Janobi 2001), texture spectrum (He and Wang 1991), Gabor filters (Grigorescu et al. 2002) and wavelets (Chang and Kuo 1993), to name a few, are powerful and continually advancing (in robustness and accuracy) tools which can be adapted to estimate deformable interfaces undergoing non-rigid motion (Kambhamettu et al. 1994). In conjunction, parametric and geometric active contours commonly known as ''snakes'' are used extensively to locate object boundaries through edge-detection in segmentation and motion tracking (Leymarie and Levine 1993; Xu and Prince 1998; Akgul and Kambhamettu 2003). In the following sections, we describe an algorithm based on GLCM and

"snakes" to calculate the complex turbulent air–water interface from PIV images of an air entraining laboratory hydraulic jump.

## 2 Experimental parameters

The PIV images used in this paper are of a laboratory generated turbulent hydraulic jump with breaking and air entrainment. The jump was set-up downstream of an undershot weir in a recirculating Armstrong S6 tilting flume that is 4.8 m long and 30 cm wide, with glass side walls 9 mm thick, and an opaque bottom. The instantaneous air–water interface in this case is inherently unsteady, with multiple time scales of fluctuations. It is also free in the sense that the interface does not support any shear. Hereafter, the interface is thus referred to as the "free surface" or simply as the "surface." The upstream to downstream depth ratio was 1.25, and the upstream Froude number was 1.2. The defining physical characteristic of the flow is the transition from supercritical to subcritical flow through energy dissipation in the jump (Henderson 1966; Rajaratnam 1965). In this particular case, the jump is marked by breaking in the transition region, leading to air entrainment not just at the toe, but also in the breaker region. In addition, the turbulent fluctuations frequently lead to ejection of water droplets into the air. The air–water interface in this two-phase region, though highly subjective, can be defined in terms of the intermittency factor (Brocchini and Peregrine 2001). The PIV set-up, shown in Fig. 1, consisted of a 120 mJ/pulse Nd-Yag New Wave solo laser source with a pulse length of 3–5 ns. This was mounted onto a custom-built, streamlined, submersible periscope which was lowered into the water. The optics in the periscope were arranged in such a way that the laser beam emerged as a planar light sheet parallel to the flume wall. The distance of the light sheet from the near side wall was 15 cm. The flume bottom and the undershot weir were painted with water resistant black marine paint to minimize reflections of the laser sheet. The water was seeded with 14 $\mu$m silver coated hollow glass spheres from Potters industries. A Kodak Megaplus 1.0 camera with a 1,016 (vertical) × 1,008 (horizontal) pixel CCD array with its image plane parallel to the (and 54.5 cm away from the near) flume wall was used to visualize the flow. A Dantec acquisition system was used to acquire the 8-bit grayscale images and store them onto a hard drive. The laser pulses were synchronized with the 30 Hz camera frame rate which ultimately led to a 15 Hz sampling rate for the instantaneous velocity fields. The time interval between the two images in a pair was 300 $\mu$s. Each experimental run consisted of an ensemble of 2,040 images (1,020 image pairs). The physical target area was 23.1 cm (horizontal) × 23.3 cm (vertical). More details of the experimental set-up and flow parameters can be found in Misra et al. (2004).

It was essential to calculate the air–water interface prior to calculating the velocities since there were spurious particle reflections near the interface arising from the perspective viewing angle of the camera. This is shown as a schematic in a two-dimensional side-view of the flume in Fig. 2. O is the center of projection. The actual interface is defined as the intersection of the laser light sheet with the water surface and is given by the point C. CD is the spuriously-imaged water surface. The

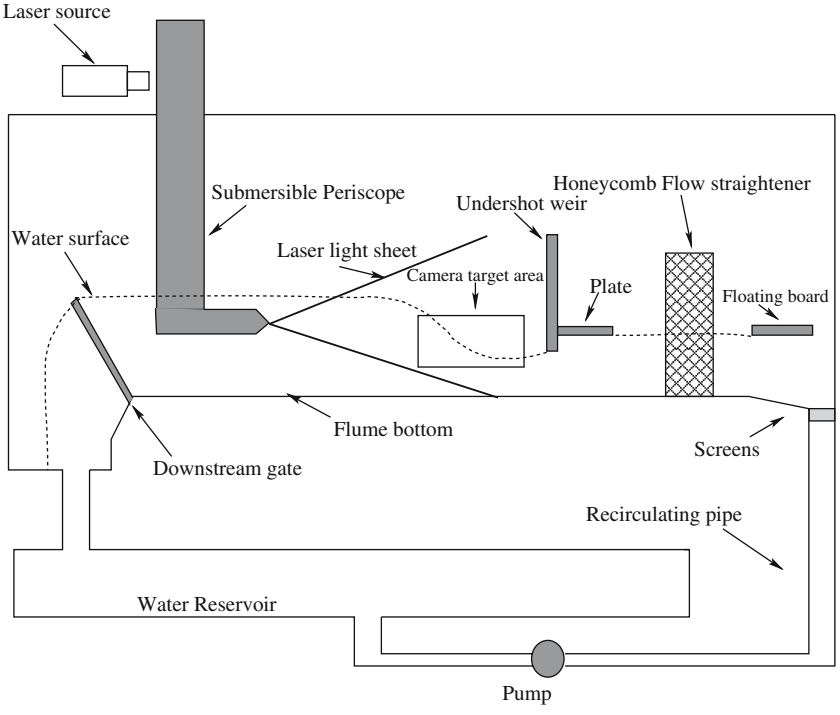


**Fig. 1** Experimental set-up

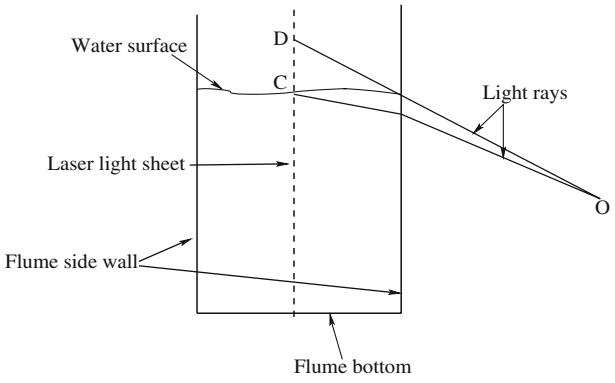


**Fig. 2** Schematic side-view of the flume showing the real and spurious air–water interfaces. See text for description

instantaneous surface itself was of interest for analyzing the intermittency of the turbulent surface fluctuations and tracking the toe of the jump (the instantaneous location of maximum curvature) in time. The detailed corrugations and break-up of the surface were not of primary interest, and in the breaker region, the surface transition from supercritical to subcritical flow is assumed to be smooth and unbroken. A typical cropped raw image is shown in Fig. 3. The flow is from right to left. The slope of the surface is nearly zero in the downstream region. The jump was therefore not undular. The reasons for a turbulent breaking jump at such a low Froude number are a combination of pre-jump entrainment, small bottom roughness and a partially developed inflow condition. The specularity seen in the raw image near the foot of the jump is from air-bubble entrainment due to turbulent shear and breaking. The physical dimensions of the cropped raw image shown in Fig. 3 are approximately 23 cm (horizontal) × 16.4 cm (vertical). Due to the rather large size of the target area, the laser light sheet attenuates from left to right, and can be clearly seen. We can see four distinct regions in the image. The intersection of the laser light sheet with the bottom divides the lower region in the image into the fluid region and the solid flume bottom. This boundary

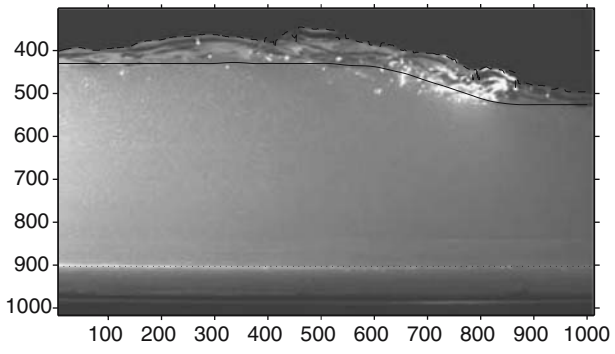


**Fig. 3** Typical raw PIV image showing spurious particle images and the interface estimates. *Dotted line* flume bottom, *dashed line* boundary between surface reflection region and air, *solid line* calculated actual free surface

is easily detected by standard edge detection techniques such as *Canny* or *Sobel* edge methods and the result is shown as the dotted line. The black region at the top of the image is air and forms a sharp boundary from the rest of the image. This boundary is the intersection of the water surface with the glass flume wall. This interface is also calculated by a *Canny* edge algorithm and is shown in the figure as the dashed line. The diffuse region between this boundary and the real surface is the image of the water surface between the laser light sheet and the flume wall. This is the region which gives a finite non-zero correlation when calculating the PIV velocities near the interface and leads, ultimately, to an artificial shear layer above the real surface. However, the illuminated seeding particles in the fluid region have a distinctive footprint. In the following, we take advantage of the particle footprint to calculate the actual air–water interface, which is the upper limit of the seeding particles dispersed in the fluid phase. Computer vision texture segmentation methods are particularly suited to this task of quantifying and classifying such image information.

## 3 Image segmentation

Gray level co-occurrence matrices (also known as gray tone spatial dependence matrices) are proven image processing tools to classify and separate the information content of an image (Tuceryan and Jain 1998). Information measures of image properties such as correlation, variance, contrast and entropy are used to calculate the inter-pixel dependence of tone and texture (Haralick et al. 1973). The GLCM, a subset of second-order structural methods, makes use of the gray level probability density function, which is computed as the conditional joint probability of pairs of pixel gray levels in a local area of the image. A selection of the appropriate quantization level is therefore essential to the level of precision of the texture classifier. The computational burden, however, increases with the level of quantization. Marceau et al. (1990) have shown that, for purposes of image classification accuracy, there is very little difference between 4 and 5-bit quantization. In addition, the statistical validity of GLCM is greatly improved for a 4-bit quantization as compared to an 8-bit quantization. A detailed discussion of quantization effects on co-occurrence texture statistics can be found in Clausi (2002). In the following, we use the 4-bit values obtained from a down-quantization of the original 8-bit grayscale data.

The main assumption behind the GLCM is that the texture-content information is represented by the average inter-pixel gray level spatial relationships. The co-occurrence of gray levels is represented mathematically by the probability, or relative frequencies of occurrence, conditional on the inter-pixel distance, $\delta$ (pixels), and the relative orientation, $\theta$ (degrees). In the following, we give a brief description of the GLCM. The relative frequency of occurrence is given by

$$C_{ij} = \frac{P_{ij}}{\sum_{i,j=1}^{N} P_{ij}}, \quad (1)$$

where $P_{ij}$ is the frequency of occurrence of gray levels pair $i$ and $j$, which are the horizontal and vertical pixel indices, respectively. The summation is over the total number of distinct gray levels in the quantized image. $N$ is therefore dependent upon the level of quantization ($m = 4$ in our case), and $N = 2^m = 16$. Once $C_{ij}$ has been calculated, the texture classes can be defined. We give three examples here, which are the texture classes used in the present algorithm. For a definition of other classes, see Haralick et al. (1973). The contrast texture is given by

$$\sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij}(i-j)^2. \quad (2)$$

The horizontal mean is defined in terms of the weighted GLCM itself,

$$\sum_{i,j=1}^{N} iC_{ij}. \quad (3)$$

The texture class called difference entropy is calculated as

$$-\sum_{k=1}^{N} p_{i-j}(k)\log\{p_{i-j}(k)\}, \quad (4)$$

where $p_{i-j}(k) = \sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij}$, $|i-j| = k$, and $k = 1, 2, \ldots, N$. The primary reason that these are chosen over other classes is because of their low degree of correlation with each other. For classification purposes, it is often advisable to choose one of the contrast measures (contrast, homogeneity, and dissimilarity), one of the orderliness measures (entropy, difference entropy, and energy) and one or more of the descriptive statistical measures (standard deviation and mean). Typically, either a linear discriminant analysis or maximum-likelihood estimation method are used to assess the contribution of several variables to the classification accuracy (Barber and LeDrew 1991). We used a simple trial-and-error method to select the best possible candidates which gave the closest fit to the visually calculated interface for a single image used for tuning the set of parameters in the algorithm. For the images considered here, a different choice of other texture classes, one from each category, gave very similar results. For more information on feature selection and pattern classification, see Weszka et al. (1976) and Duda et al. (2000).

## 4 Interface estimation

### 4.1 Pre-processing

Before we calculate the GLCM, we pre-process the image to make it a more robust candidate for the texture segmentation algorithm. To save computation time, only a section of the raw image spanning the diffuse region is analyzed. The section (hereafter referred to as "image") size is 300 pixels (vertical) × 1,008 pixels (horizontal), spanning the interface. We first calculate the magnitude of the gradient of this image

$$I_m = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}, \quad (5)$$

where $I(x, y)$ is the grayscale intensity, and $x$ and $y$ are pixel coordinates in the horizontal and vertical directions. The seeding particles show up as cross-shaped features in the gradient image, and can be clearly seen in a zoomed-in region shown in Fig. 4. The structure of these features shows that the particle image footprint is approximately equal to 1 pixel, which can be verified using the formula for particle image diameter (Prasad 2000),

$$d_\tau = \sqrt{M^2 d_p^2 + [2.444(1+M)f^{\#}\lambda]^2}, \quad (6)$$

where $d_\tau$ is the particle image diameter, $M$ is the magnification ($= 0.04$), $d_p$ is the particle diameter ($= 14$ $\mu$m), $f^{\#}$ is the $f$-number of the lens ($= 6$) and $\lambda$ is the wavelength of the laser ($= 532$ nm). Since the pixel spacing on the CCD is $d_{pix} = 9$ $\mu$m, we get $(d_\tau/d_{pix}) = 0.9$.

Based on the particle footprint, a template for a particle is chosen in the form

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

To enhance the contribution of the real seeding particles in the image, the template is then cross-correlated with the gradient magnitude image and then multiplied with the raw image to yield a "weighted" image, shown in Fig. 5. The fluid region with the particles can be seen separated from the diffuse reflection region and the air above.

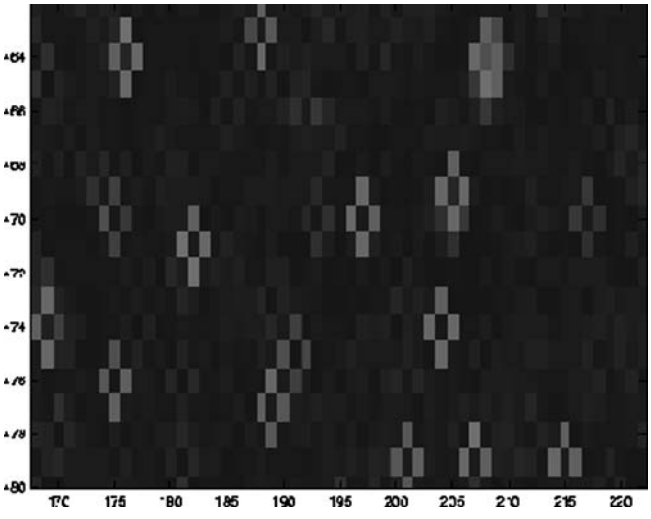


**Fig. 4** Seeding particle footprint in gradient image

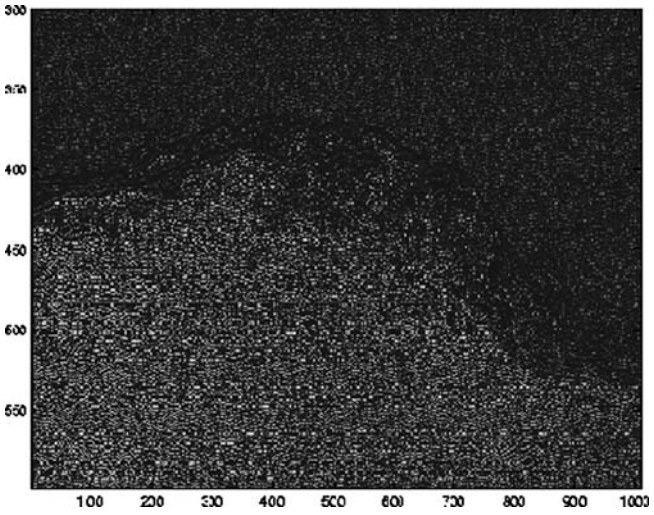


**Fig. 5** Weighted image

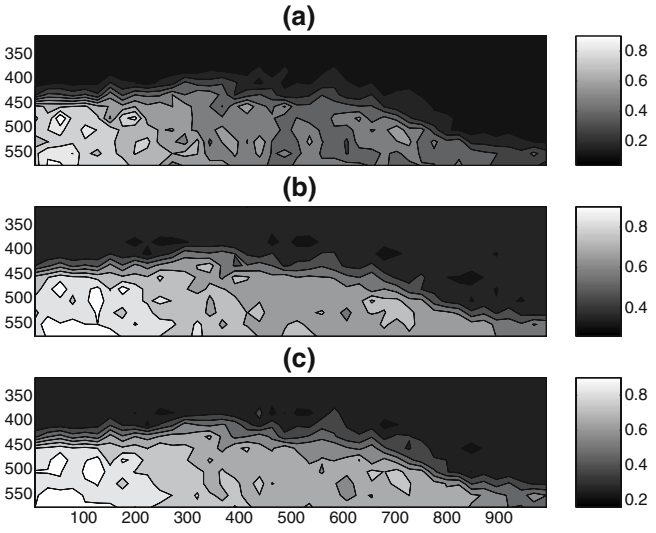


**Fig. 6** Normalized texture classes, **a** contrast, **b** horizontal mean, and **c** difference entropy

## 4.2 Gray level co-occurrence matrix

The next step is to choose a window size for calculating the GLCM since the texture statistic is computed as a single measure of the total information content. The size of the window is a trade-off between the robustness of the estimate (directly proportional to the window size) and the resolution (inversely proportional to window size). We found a window of 24×24 pixels, with no overlap, to work reasonably well. We note that, even though simple to implement and increased resolution can be obtained by overlapping in the vertical direction, the computational burden increases with the extent of overlap. We use a displacement of 3 pixels between the reference and neighbor pixels, i.e., $|\delta| = 3$. All four possible relative orientations are take into account by western ($\theta = 0$), southern ($\theta = 90$), northwestern ($\theta = 135$) and southwestern ($\theta = -135$) offsets. The opposite orientations are accounted for when constructing the symmetric GLCM. The raw GLCM is first formed by summing the frequencies of co-occurrence of gray level values, over all possible pixel pairs given by the range of $\delta$ and $\theta$ values in the window. The symmetric GLCM is calculated by adding the transpose of the raw GLCM to itself. It is therefore assumed that the pixel pair gray scale values have no preferred orientation. This is intuitively justified, since the seeding particles are distributed randomly in the fluid phase. The symmetric GLCM is then normalized by its sum, thereby yielding the relative frequencies of occurrence. The texture classes (horizontal mean, contrast and difference entropy) are then calculated, and are shown in Fig. 6. There is, surprisingly, a remarkable qualitative similarity between them. Considering that these particular texture classes are not well-correlated by definition, the similarity is therefore an indication of the comparable degree of representation of the image content in each of the texture classes.

## 4.3 Singular value decomposition

It has been shown that the information contained within the GLCM is not fully represented by any single texture class (Conners and Harlow 1980). With that in mind, the texture classes are combined into a single matrix by performing principal component analysis on the texture class covariance matrix (Turk and Pentland 1991). Each texture class is thus collected together to form $\mathbf{A} = [T_1, T_2, \ldots, T_M]$, where $T_i$ are the individual texture classes and each row of $\mathbf{A}$ corresponds to a collocated position from each of the classes. The texture covariance matrix can be computed as $\mathbf{C} = \Phi\Phi^{\mathrm{T}}$, where $\Phi = \mathbf{A} - \bar{\mathbf{A}}$ with $\bar{\mathbf{A}}$ being the average texture. A single matrix can be calculated by projecting the texture class data points along $K$ ($K \leq M$) orthonormal principal vectors obtained by eigenvalue decomposition of the texture covariance matrix. In our computations, we observed that the first eigenvalue–eigenvector pair provided sufficient robustness to the subsequent stages of the process.

The edge map of the singular value decomposition matrix is calculated using a *Canny* edge detection method. The edge map is then blurred using a Gaussian convolution. It may be less clear why there is any interest in blurring the edge map. It does, however, perform a useful intermediate step, and is commonly used in edge-detection routines in image processing (Gonzales and Woods 2002). A Gaussian blur is a general purpose low-pass blur filter. This removes fine image detail and noise inherent to an edge detection method (such as *Canny*), leaving only larger scale changes. Gaussian blurs therefore produce a smoothing effect without side effects. Obviously, the level of detail retained depends on the size of the filter, which is given by the standard deviation. In the present, the Gaussian kernel selected is 3×3 pixels with a standard deviation of 1 pixel. The "Laplacian of the Gaussian" operator can

also be used to do both smoothing and edge detection operations at once (Marr and Hildreth 1980).

### 4.4 Active contours

The next step is to determine how to represent the set of (first approximation) discrete points obtained from the gradient of the Gaussian blurred image, which have been assumed to lie on an edge, which in our case is the turbulent interface. Kass et al. (1987) were the first to propose a model called active contours or "snakes" in which the final form of the contour is influenced by a feedback from a higher level process essentially minimizing the "energy." "Snakes" were previously used by Duncan et al. (1999) to calculate the crest profile evolution for gentle spilling breakers. Hermite cubic polynomials were used to represent the "snakes." The experimental and camera set-up were, however, tuned to solely measure the scalar interface from the dye visualization, and the images analyzed are therefore closer to typical LIF images than PIV images. According to the classical definition, the snake is a deformable contour $v(\mathbf{s}) = (x(s), y(s))$ parameterized by the arc-length $s \in [0, 1]$. The contour is controlled by balancing the internal energy terms ($E_{\text{int}}$) emanating from the shape of the contour, with the external energy ($E_{\text{ext}}$) extracted from the image and/or obtained from a higher level image understanding processes (such as the texture information provided by the GLCM). The energy functional to be minimized takes the form,

$$E^* = \int_0^1 [\alpha|v'(s)|^2 + \beta|v''(s)| + \gamma E_{\text{ext}}]\mathrm{d}s. \tag{7}$$

Primes denote differentiation with respect to $s$. The first two terms form the internal energy while the last term denotes the external energy. The internal energy is responsible for the smoothness of the contour while the external energy attracts the contour towards the object boundary. The parameters $\alpha$, $\beta$, and $\gamma$ control the importance of each term towards the net energy functional. There are many external energy functionals possible, with the image gradient $E_{\text{ext}} = -|\nabla\ I(v(s))|$ being the most common choice.

The discrete approximation of the active contour renders the minimization of the energy functionals over $N$ discrete samples of $v(s)$, also known as snaxels ($v_i = [x_i, y_i]$, $i = 1, \ldots, N$).

$$E^* = \sum_{i=1}^{N} [\alpha E_{\text{cont}}(v_i) + \beta E_{\text{curv}}(v_i) + \gamma E_{\text{ext}}(v_i)]. \tag{8}$$

The derivatives present in the internal energy terms can be replaced by the discrete counterparts as $E_{\text{cont}}(v_i) = (d_{\text{avg}} - ||v_i - v_{i-1}||)^2$ which is a first-order continuity constraint similar to the first-derivative term in Kass et al. (1987), and seeks to keep points spaced equally along the curve. $E_{\text{curv}}(v_i) = ||v_{i+1} - 2v_i + v_{i-1}||^2$ is analogous to a second derivative term which seeks to minimize the curvature. $d_{\text{avg}}$ is defined as the average distance between the contour points and is required so that the contour does not shrink towards itself. Note that each term has to be normalized by dividing by the largest value of that term within the neighborhood of the search. In our algorithm, the external energy that is used to control the active contour is based on the data obtained from the texture classes. We thus try to minimize the magnitude of the gradient of the texture classes between the seeding particle zone and the non-particle zone under the constraint that the contour so obtained satisfies first and second-order continuity. The contour estimation is repeated, with the estimate at one stage forming the initialization to the subsequent stages, until the contour stabilizes over the texture classes. In the implementation, we have used the "greedy" algorithm described by Williams and Shah (1992) instead of the classical formulation. Here, a local search is performed at the position of each discrete point and the position that minimizes the energy functional is selected as the new updated position. The local search neighborhood is maintained at 15 vertical pixels (seven on either side) at each iteration. Finally, a cubic spline is used to interpolate the discrete snaxels and to obtain a continuous contour giving the free surface estimate.

A single randomly selected image is used to tune the parameters discussed above by comparison with the visually-interpreted (hereafter, VI) surface (see 1) for that image. The rest of the images in the ensemble are batch processed without any further intervention. Nonetheless, to test the algorithm, we selected random images, and on comparing the GLCM surface with the VI surfaces, found good agreement between the two for the fixed set of parameters.

## 5 Results

Since the flow is very susceptible to external perturbations, no intrusive probes were used to either measure the instantaneous surface or the flow velocities. In-situ measurements are therefore not available as ground truth. There are, however, several ways to test the accuracy of the instantaneous as well as ensemble averaged calculated interfaces and are described below.

### 5.1 Visual comparison

The results from the GLCM algorithm were assessed first with respect to human perception. It is noted that such a comparison does not provide an objective verification of the accuracy of the estimated interface, but in the absence of ground-truth measurements, in the vision

research community, the results from image-processing algorithms are often compared, and often benchmarked, with estimates of visual-interpretation to test the accuracy of segmentation algorithms (Malik et al. 2001; Martin et al. 2001; Mojsilović et al. 2002; Li et al. 2003; Rasmussen 2004; O'Callaghan and Bull 2005). Ten randomly selected (from the ensemble) PIV images were distributed to each of a group of ten participants. Of the ten subjects, two had prior experience with image-processing methods in the context of PIV. A short problem description was provided and the subjects were asked to click on the image where they perceived the position of the true interface which was assumed to be the upper limit of the seeding particles. Since the inputs from the subjects could be sparse and not coincident with the GLCM calculated positions, a cubic spline was used to interpolate the VI surface at the GLCM locations. The mean position of the contour was then computed as the average of the contours obtained from the ten participants and is shown as the dotted line in Figs. 7 and 8 for two randomly selected images (we will call them A and B) in the ensemble. The error bars show the variation between the individual visual estimates. The GLCM interface is shown as the solid line. It is clearly seen that the subjectivity in the visual interpretation increases in the specular breaking region. The overall qualitative agreement between the average VI interface and the GLCM interface is good, even in the breaker region, where the surface is broken up and there is two-phase flow resulting from ejection of water particles into the air and entrainment of air bubbles in the water. As noted earlier, we have assumed an unbroken smooth contour through the breaker region. The quantitative disagreement between the GLCM and VI surfaces is shown in Fig. 9 and is given by

$$\epsilon(x) = h(x)_{\mathrm{VI}} - h(x)_{\mathrm{GLCM}}. \tag{9}$$

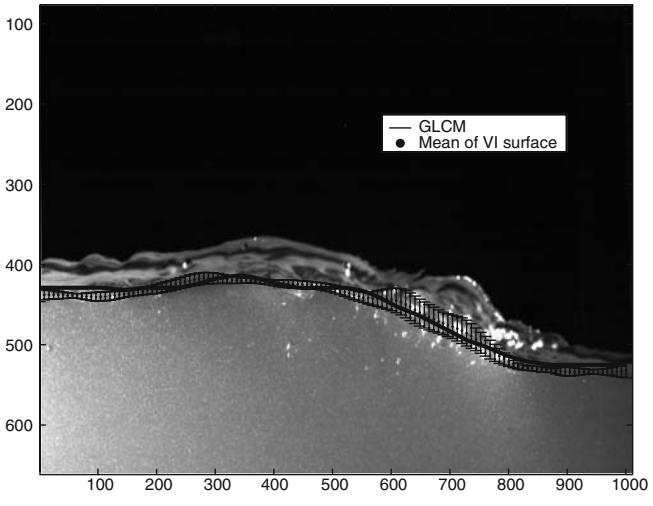


**Fig. 7** Comparison of GLCM and visually calculated interfaces for a typical image (*A*)

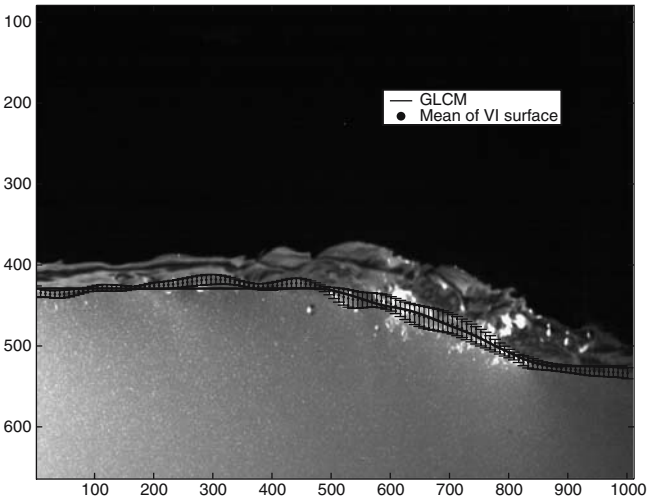


**Fig. 8** Comparison of GLCM and visually calculated interfaces for a typical image (*B*)

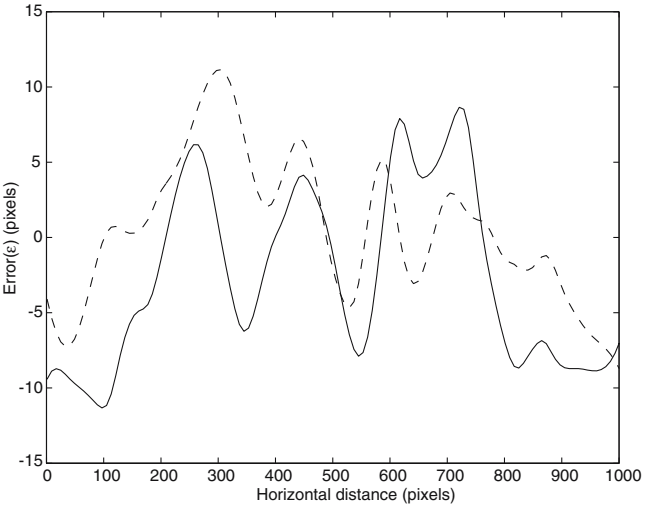


**Fig. 9** Error ($\varepsilon$) between GLCM and mean VI surfaces for image A (*solid line*) and image B (*dashed line*)

$h(x)$ is the local water depth. Negative values of $\varepsilon$ therefore imply that the GLCM surface is above the VI surface and positive values indicate that it is below. A first qualitative conclusion which can be drawn is that there is no consistent bias towards under- or over-prediction by the GLCM method, which is reassuring. One of the regions where there are large errors for both images is between 500 and 800 pixels. This is the region in the image occupied by the specularity resulting from air-entrainment due to breaking. In this region, it is even physically impossible to define a continuous interface since there is ejection of water droplets along with entrainment of air bubbles. Hence, neither the GLCM nor the VI interface provide the right answer. However, the subjects were instructed to assume an unbroken and smooth interface, as was assumed by the GLCM algorithm. The second region where there are large errors is

near the far end (upstream end) of the domain, where the GLCM interface is higher than the VI interface. In this region, due to the laser attenuation resulting from the rather large target area, there is very little information to visually distinguish between the image imprint of a particle and its spurious reflection, which is indirectly evident from the decreasing values of the texture classes in Fig. 6. Therefore, the GLCM is not as accurate in this region. It is noted that any other image processing method would probably fail in this region as well. The errors in the near end (downstream end) of the domain for both images is because of the assumption of a small slope in the GLCM interface in this region, which was enforced during the tuning of the parameters. This was justified because the jump was not undular, as can be seen from the small slopes of the VI interfaces as well. The smaller perturbations of the interface were not of interest. However, there is no bias in the comparison, since comparisons in other images do show a positive error of similar magnitudes in this region. The mean (averaged in the horizontal direction) absolute errors for the two images are $\bar{\epsilon}_A = 5.77$ pixels ($=1.3$ mm) and $\bar{\epsilon}_B = 3.85$ pixels ($=0.9$ mm), which were within the measurement location resolution for the calculation of the PIV velocities ($=8$ pixels). The mean non-dimensionalized (by the horizontally-averaged GLCM depth) absolute error percentages are $\bar{\epsilon}'_A = 1.31\%$ and $\bar{\epsilon}'_A = 0.87\%$. As to computational efficiency, on a personal computer, with a 3 GHz Intel processor and 1.5 Gb of physical memory, the GLCM surface is calculated for a single image in approximately 15 s.

### 5.2 Volume flux conservation

On ensemble averaging the instantaneous surfaces, the upstream and downstream water depths were calculated to be $h_0 = 8.62$ cm and $h_1 = 10.85$ cm, respectively, which agreed well with visually observed (when performing the experiments) instantaneous depths of 8.5 and 10.9 cm, respectively. Using the calculated depths from Belanger's equation,

$$\mathrm{Fr} = \sqrt{\frac{1}{2}\frac{h_1}{h_0}\left(\frac{h_1}{h_0}+1\right)}, \tag{10}$$

the upstream Froude number $\mathrm{Fr} = 1.1932$. Once the instantaneous surface was calculated, the region above it was padded with zeros before calculating the velocities with the PIV algorithm (Thomas et al. 2005). As mentioned before, an over-estimation of the surface would lead to the inclusion of false image information (which leads ultimately to a false shear layer and reduction in the computed velocities there) and an under-estimation of the surface would lead to an under prediction of the vertically integrated volume flux. A check on the accuracy of the vertical profiles of the velocities at various horizontal locations, and hence, indirectly, a check on the location of the instantaneous and ensemble averaged surfaces, can therefore be made in two ways. Once the ensemble averaged velocities have been calculated, we can compute the mean volume flux as,

$$Q(x) = \int_{y_1}^{h(x)} U(x,y)\mathrm{d}y, \tag{11}$$

where $x$ is the horizontal coordinate, $y$ the vertical coordinate (measured upward from the flume bottom) and the origin is chosen at an arbitrary location upstream of the jump. Note that the lower limit of integration is $y_1$, which is 16 pixels away from the flume bottom. The contribution to the flux near the bottom, including the boundary layer, cannot be estimated due to lack of data. No attempts were made to extrapolate the velocities to the bottom by fitting analytic profiles. The standard deviation ($Q_\sigma$) and mean ($\bar{Q}$) fluxes are calculated to be 0.5682 and 21.6646 l/s, which gives $Q_\sigma/\bar{Q} = 0.262$. Similar error estimates were found for low Froude number jumps from the laser Doppler velocimetry data of Bakunin (1995). Note that in addition to the bottom boundary layer, there are cross-channel variations in the flow and side-wall boundary layers, the effects of which are not accounted for, and which further reduce the deviation in the volume flux (Svendsen et al. 2000). The mass conservation was also checked locally by numerically calculating the divergence of the mean velocity field to second-order accuracy, and was found to be within measurement and digitization errors throughout the domain. This leads us to believe that the vertical profiles of the velocities, and therefore, the free surface location, in an ensemble-averaged sense, are calculated accurately. We note that this is not meant to replace a more objective verification of the instantaneous location of the interface such as would be obtained from a wave-gage.

### 5.3 Intermittency factor, curvature and slope of surface

Corrsin and Kistler (1955) have shown that the inferred distribution of turbulent free surface displacements closely follow a Gaussian error law distribution,

$$p(h) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{1}{2}\left(\frac{y-\langle h\rangle}{\sigma}\right)^2\right], \tag{12}$$

where $\langle . \rangle$ denotes an ensemble average. $p(h)$ and $\sigma$ are, respectively, the probability density function and the standard deviation for the displacement $h$. The intermittency factor is related to $p(h)$ as

$$\gamma(x,y) = \int_{y}^{\infty} p(h,x)\mathrm{d}h, \tag{13}$$

and leads to an error function profile (Brocchini and Peregrine 2001),

$$\gamma(x,y) = \frac{1}{2}\left[1 - \mathrm{erf}\left(\frac{h - \langle h\rangle}{\sqrt{2}\sigma}\right)\right]. \quad (14)$$

Figure 10 shows the calculated intermittency factor compared to the error function profile at various locations inside the breaker region. The foot of the jump (taken as the ensemble averaged location of the maximum free surface curvature) is located at $(x/h_0) \sim 0.5$. The calculated values fit the profiles well, with accuracy increasing in the downstream direction. This shows that the unsteady fluctuations of the interface, captured well by an automated GLCM algorithm, follow Gaussian error law distribution.

Another important consequential benefit of the interface estimation process is the calculation of the slope and curvature from the free surface profile. The curvature is defined as

$$\kappa = \frac{\frac{\mathrm{d}^2 h}{\mathrm{d}x^2}}{\left[1 + \frac{\mathrm{d}h}{\mathrm{d}x}^2\right]^{3/2}}. \quad (15)$$

The toe of the jump can be defined as the location of the maximum curvature and by tracking its location in time, the unsteadiness of the toe motion can be estimated since it is known that in quasi-steady and steady breakers are accompanied by coherent toe oscillations (Duncan 2001). In the present experiment, the toe oscillation was unfortunately masked by a 20-s oscillation in the flume. This was independently verified from two Sontek acoustic Doppler velocimeters placed near the inflow and outflow sections of the flume, 200 cm upstream and 175 cm downstream of the weir, respectively. The slope of the ensemble averaged surface at the location of the foot of the breaker (defined by Brocchini and Peregrine (2001) as the mean position of the toe of the breaker) was found to be 11°. This is close to the stable equilibrium value (10°) typically used for Boussinesq modeling of broken waves (Schaffer et al. 1993) and is also in good agreement with Longuet-Higgins (1973), who, in a theoretical model of flow separation at a free surface, showed that immediately upstream of the toe, the free surface should be inclined to the horizontal at an angle lying between 10 and 20°. Again, the reader is cautioned that even though the above results support both the spatial and temporal robustness of the estimates interfaces, a more direct and objective verification of the accuracy of the instantaneous interface location can only be made with measurement gages.

## 6 Conclusions

Particle image velocimetry images of an air-entraining turbulent hydraulic jump generated in a laboratory are analyzed to estimate the instantaneous complex

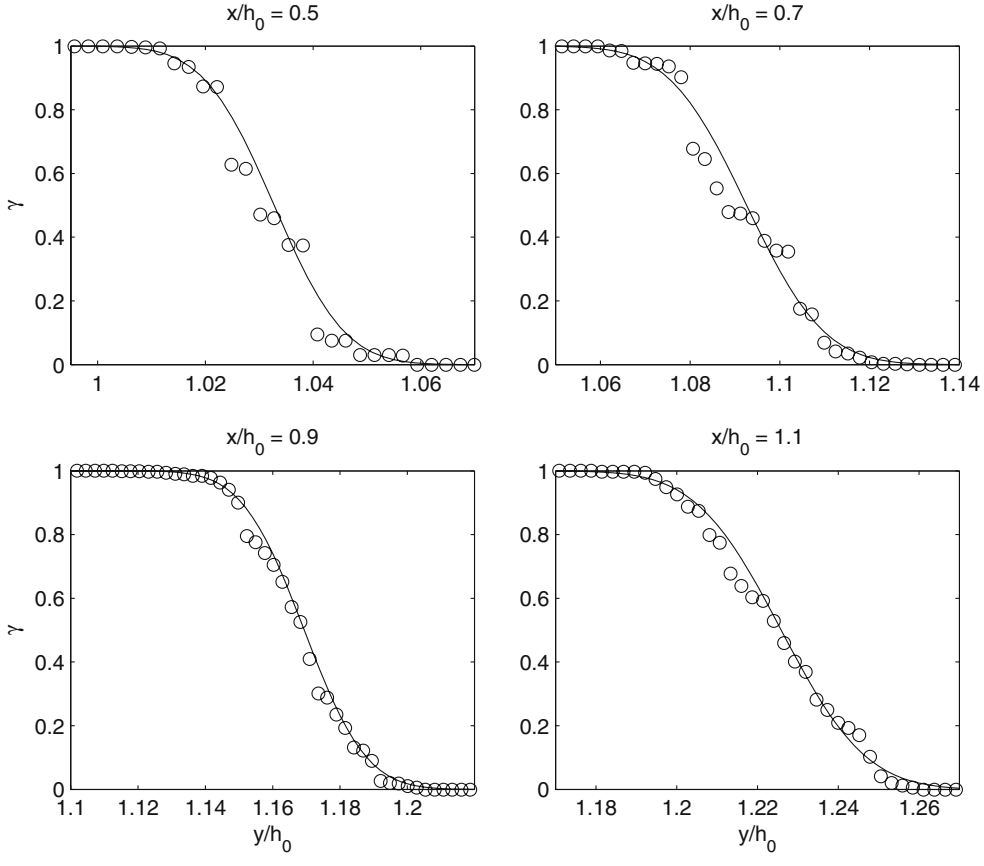


**Fig. 10** Intermittency factor $\gamma$ in the breaker region: measured data (*circle*), error function profile from Eq. 14 (*line*)

air–water interfaces using image processing methods. The algorithm presented in this paper, a second-order statistical method based on GLCM, is simple to implement and gives a first approximation to the location of the interface. Several pre-processing steps are introduced to increase the robustness of the candidate for the GLCM. A singular value decomposition is performed on the combined selected texture classes to initialize the edge map for the subsequent contour mapping process. Active contours minimizing energy functionals based on the internal (controlling the shape of the contour) and external (characterizing image information) energies of the contour are used to increase the accuracy of the predicted interface. A disadvantage with the algorithm is that the texture measures employed and the parameter values chosen for the contour extraction are obtained from an empirical evaluation. For typical PIV applications, this is a very small price to pay in light of the voluminous amount of image data generated, since a single image is used to tune the algorithm and the rest of the ensemble is analyzed in an accurate and computationally efficient manner. This process is completely automated.

In the turbulent hydraulic jump experiment analyzed here, the free air–water interface is compared to the visually-interpreted interface and the deviation is found to be within the (typical) measurement resolution (=8 pixels) for the fluid velocities. For the two images shown, the horizontally-averaged absolute errors were 5.77 and 3.85 pixels. An accurate estimation of the surface enabled the exclusion of spurious interfacial reflections in the images and ensured volume flux conservation throughout the domain. The intermittency factor representing the turbulent fluctuations of the interface was seen to fit theoretical profiles well, with accuracy increasing in the downstream direction. The geometrical slope of the ensemble averaged surface near the foot of the breaker agreed with the value used in numerical simulation of saturated spilling breakers and a theoretical model of flow separation at a free surface.

**Acknowledgements** This work was supported by the National Oceanographic Partnership Program (NOPP), grant N00014-99-1-1051.

## 7 Appendix

For the experiment described in the main body of the paper in Sect. 2, a separate dataset was collected from an optically zoomed-in target area in the breaker region to study the detailed turbulence structure of the roller region and the breaker shear layer. The physical dimensions were 11.09×11.18 $cm^2$. The surfaces were estimated with the same GLCM and snake parameters described in the paper and the VI surface is compared to the GLCM surface in Fig. 11. The error bars denote one standard deviation of the VI surface. The horizontally-averaged absolute error, $\bar{\epsilon}$ in this case is

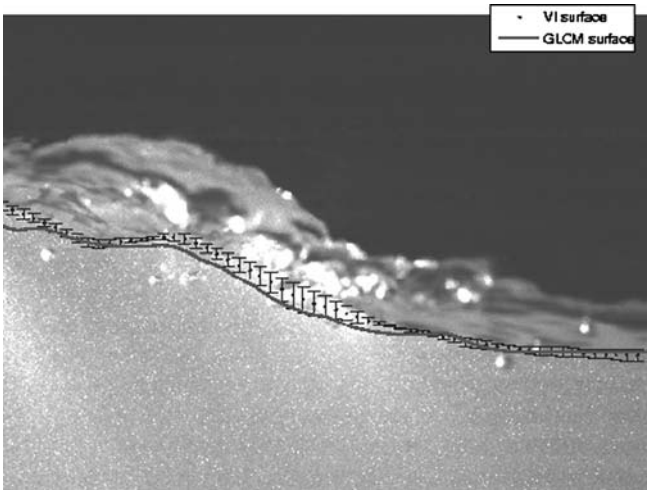


**Fig. 11** Comparison of GLCM and visually calculated interfaces for a typical image in the zoomed-in dataset

7.9 pixels (=0.8 mm). In the zoomed-in dataset, the specularity in the breaking region occupies a larger (approximately twice) portion of the image. In this region, as noted before, the definition of a smooth unbroken interface is highly subjective and this can be seen in the increasingly wide error bars in that region. We note, however, that the GLCM does a very reasonable job of estimating most of the small scale perturbations of the surface even in this two-phase breaker region.

## References

Akgul YS, Kambhamettu C (2003) A coarse-to-fine deformable contour optimization framework. IEEE Trans Pattern Anal Mach Intell 25(2):177–186

Al-Janobi A (2001) Performance evaluation of cross-diagonal texture matrix method of texture analysis. Pattern Recognit 34:171–180

Bakunin J (1995) Experimental study of hydraulic jumps in low Froude number. Master's Thesis, University of Delaware

Barber DG, LeDrew EF (1991) SAR sea ice discrimination using texture statistics: a multivariate approach. Photo Eng Rem Sens 57(4):385–395

Brocchini M, Peregrine DH (2001) The dynamics of strong turbulence at free surfaces. Part 1. Description. J Fluid Mech 449:225–254

Chang T, Kuo C (1993) Texture analysis and classification with tree-structured wavelet transform. IEEE Trans Image Proc 2(4):429–441

Clausi DA (2002) An analysis of co-occurrence texture statistics as a function of grey-level quantization. Can J Rem Sens 28(1):45–62

Conners RW, Harlow CA (1980) A theoretical comparison of texture algorithms. IEEE Trans Pattern Anal Mach Intell 2(3):204–222

Corrsin S, Kistler AL (1955) Free-stream boundaries of turbulent flows. Technical Report NACA, Washington, p 1244

Dabiri D, Gharib M (1997) Experimental investigation of vorticity generation within a spilling water wave. J Fluid Mech 330:113–129

Dewey CF (1976) Qualitative and quantitative flow field visualization using laser-induced fluorescence. In: Proceedings of the AGARD conference on application of non-intrusive instrumentation in fluid flow research AGARD-CP-193(17), pp 1–7

Donelan MA, Haus BK, Reul N, Plant WJ, Stiassnie M, Graber HC, Brown OB, Saltzman ES (2004) On the limiting aerodynamic roughness of the ocean in very strong winds. Geophys Res Lett 31:L18306

Duda RO, Hart PE, Stork DG (2000) Pattern classification, 2nd edn. Wiley, New York

Duncan JH (2001) Spilling breakers. Ann Rev Fluid Mech 33:519–547

Duncan JH, Qiao H, Philomin V, Wenz A (1999) Gentle spilling breakers: crest profile evolution. J Fluid Mech 379:191–222

Gonzales RC, Woods RE (2002) Digital image processing, 2nd edn. Prentice Hall, Englewood cliffs

Grigorescu SE, Petkov N, Kruizinga P (2002) Comparison of texture features based on Gabor filters. IEEE Trans Image Proc 11(10):1160–1167

Haralick RM, Shanmugam K, Dinstein I (1973) Texture parameters for image classification. IEEE Trans Syst Man Cybern 3(6):610–621

Hassan YA, Okamoto K, Philip OG (1996) Investigation of the interaction between a fluid flow and the fluid's free surface using particle image velocimetry. In: 9th International symposium on transport phenomena in thermal-fluids engineering ISTP, vol 9, pp 566–574

He D, Wang L (1991) Texture features based on texture spectrum. Pattern Recognit 24(5):391–399

Henderson FM (1966) Open channel flow. The MacMillan Company, New York

Hirsa AH, Vogel MJ, Gayton JD (2001) Digital particle velocimetry technique for free-surface boundary layer measurements: application to vortex pair interactions. Exp Fluids 31:127–139

Kambhamettu C, Goldgof DB, Terzopoulos D, Huang TS (1994) Nonrigid motion analysis. In: Young TY (ed) Handbook of pattern recognition and image processing: computer vision. Academic, San Diego, pp 405–430

Kass M, Witkin A, Terzopoulos D (1987) Snakes: active contour models. Int J Comp Vis 1:321–331

Law CNS, Khoo BC, Chen TC (1999) Turbulence structure in the immediate vicinity of the shear-free air–water interface induced by a deeply submerged jet. Exp Fluids 27:321–331

Leymarie F, Levine MD (1993) Tracking deformable objects in the plane using an active contour model. IEEE Trans Pattern Anal Mach Intell 15(6):617–634

Li M, Kambhamettu C, Stone M (2003) Snake for band edge extraction and its applications. In: 6th IASTED international conference on computer graphics and imaging, Hawaii, pp 212–217

Lin HJ, Perlin M (1998) Improved methods for thin, surface boundary layer investigations. Exp Fluids 25:431–444

Longuet-Higgins MS (1973) A model of flow separation at a free surface. J Fluid Mech 57(1):129–148

Malik JA, Belongie S, Leung T, Shi J (2001) Contour and texture analysis for image segmentation. Intl J Comp Vis 43(1):7–27

Marceau DJ, Howarth PJ, Dubois J-M, Gratton DJ (1990) Evaluation of the grey-level co-occurrence matrix method for land-cover classification using SPOT imagery. IEEE Trans Geosci Rem Sens 28(4):513–519

Marr D, Hildreth E (1980) Theory of edge detection. Proc R Soc Lond B 207:187–217

Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of the international conference on computer vision, pp 416–423

Misra SK, Kirby JT, Brocchini M, Thomas M, Veron F, Kambhamettu C (2004) Extra strain rates in spilling breaking waves. In: Proceedings of the 29th ICCE, Lisbon, vol 1, pp 370–378

Mojsilović A, Hu J, Soljanin E (2002) Extraction of perceptually important colors and similarity measurement from image matching, retreival, and analysis. IEEE Trans Imag Proc 11(11):1238–1248

O'Callaghan RJ, Bull DR (2005) Combined morphological-spectral unsupervised image segmentation. IEEE Trans Imag Proc 14(1):49–62

Okamoto K, Philip OG, Hassan YA (1996) Study on the interaction between free surface and flow using PIV. In: Chen C, Shih C, Lienau J, Kung RJ (eds) Flow modeling and turbulence measurements VI. Balkema, Rotterdam, pp 185–192

Peirson WL (1997) Measurement of surface velocities and shears at a wave air–water interface using particle image velocimetry. Exp Fluids 23:427–437

Prasad AK (2000) Particle image velocimetry. Curr Sci 79(1):51–60

Prasad RR, Sreenivasan KR (1989) Scalar interfaces in digital images of turbulent flows. Exp Fluids 7(4):259–264

Qiao H, Duncan JH (2001) Gentle spilling breakers: crest flow-field evolution. J Fluid Mech 439:57–85

Raffel M, Willert C, Kompenhans J (2001) Particle image velocimetry: a practical guide. Springer, GmbH & Co., Berlin Heidelberg, pp 207–248

Rajaratnam N (1965) The hydraulic jump as a wall jet. J Hyd Div Proc Am Soc Civil Eng 5(4482):107–132

Rasmussen C (2004) Grouping dominant orientations for ill-structured road following. In: IEEE conference CVPR-04, Washington D.C.

Schaffer HA, Madsen PA, Deigaard R (1993) A Boussinesq model for waves breaking in shallow water. Coastal Eng 20:185–202

Svendsen IA, Veeramony J, Bakunin J, Kirby JT (2000) The flow in weak turbulent hydraulic jumps. J Fluid Mech 418:25–57

Thomas M, Misra SK, Kambhamettu C, Kirby JT (2005) A robust motion estimation algorithm for PIV. Meas Sci Technol 16(3):865–877

Tsuei L, Savas Ö (2000) Treatment of interfaces in particle image velocimetry. Exp Fluids 29:203–214

Tuceryan M, Jain AK (1998) The handbook of pattern recognition and computer vision. World Scientific, pp 207–248

Turk M, Pentland A (1991) Eigenfaces for recognition. J Cognit Neurosci 3:72–86

Westerweel J, Hofmann T, Fukushima C, Hunt JCR (2002) The turbulent/non-turbulent interface at the outer boundary of a self-similar turbulent jet. Exp Fluids 33:873–878

Weszka JS, Dyer CR, Rosenfeld A (1976) A comparative study of texture measures for terrain classification. IEEE Trans Syst Man Cybern 6(4):269–285

Williams DJ, Shah M (1992) A fast algorithm for active contours and curvature estimation. CVGIP Image Understanding 55(1):14–26

Xu C, Prince JL (1998) Snakes, shapes, and gradient vector flow. IEEE Trans Image Proc 7(3):359–369