

## Image based flow visualisation of experimental flow fields inside a gross pollutant trap

J.T. Madhani<sup>1</sup>, J. Young<sup>2</sup> and R.J. Brown<sup>1</sup>

<sup>1</sup> School of Chemistry, Physics and Mechanical Engineering, Faculty of Science and Technology, Queensland University of Technology, Brisbane, QLD 4000, Australia

<sup>2</sup> HPC and Research Support Group, Queensland University of Technology, Brisbane, QLD 4000,

### Abstract

Typical flow fields in a stormwater gross pollutant trap (GPT) with blocked retaining screens were experimentally captured and visualised. Particle image velocimetry (PIV) software was used to capture the flow field data by tracking neutrally buoyant particles with a high speed camera. A technique was developed to apply the Image Based Flow Visualization (IBFV) algorithm to the experimental raw dataset generated by the PIV software. The dataset consisted of scattered 2D point velocity vectors and the IBFV visualisation facilitates flow feature characterisation within the GPT. The flow features played a pivotal role in understanding gross pollutant capture and retention within the GPT. It was found that the IBFV animations revealed otherwise unnoticed flow features and experimental artefacts. For example, a circular tracer marker in the IBFV program visually highlighted streamlines to investigate specific areas and identify the flow features within the GPT.

### Introduction

Hydrodynamic characteristics of waste and stormwater holding devices, such as gross pollutant traps (GPTs), provide valuable insights into pollutant capture and retention characteristics owing to regions of flow recirculation and critical (high and low) velocities. Field studies have shown that internal screens in GPTs are often blocked due to infrequent cleaning [1]. Blocked screens and low inlet flows can change radically the hydrodynamic structure and the stormwater pollutant capture/retention characteristics of a GPT.

The hydrodynamic characteristics of a fully blocked GPT with a low flow inlet were initially investigated using single point velocity measurements [2]. To experimentally capture and visualise an extensive set of flow field data, an image based vector visualisation method—the Line Integral Convolution (LIC)—was implemented in an earlier work [1]. In [1] it was concluded that the LIC method was superior to point-based or discrete object visualisation (e.g., hedgehog or arrows plots as produced by the PIV software) in terms of conveying information about every point within a domain. While [1] acknowledged some uncertainties in the 2D depth coverage, the simplified approach permitted the added benefit of collecting extensive flow field data, which otherwise would be labour intensive.

For this study, the visualisation was extended to include the animation of low flow through GPT with fully blocked screens. The dataset was collected using particle image velocimeter (PIV) software to track neutral buoyant particles in an experimental GPT rig with a high speed camera. A technique was developed to apply IBFV to the captured non-uniform, scattered, statistical mean point velocities. Comparisons were made between the visualised dataset using the IBFV and the LIC methods for an inlet flow condition. The flow features were also compared with the previously defined CFD predictions along with the deposition

of sediments and artificial gross pollutants obtained from capture/retention experiments [3].

### Experimental overview

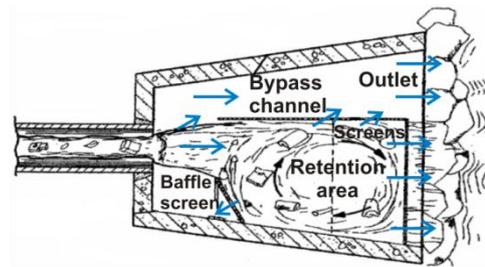


Figure 1. Plan view of the *LitterBank* gross pollutant trap.

Figure 1 shows a plan view of the investigated GPT, the *LitterBank*,—recently developed by C-M Concrete Pty Ltd.—which is operating in several locations throughout Queensland, Australia. In ideal operating conditions, the upstream stormwater directly enters the retention area (litter trap) of the device via the inlet. Here, the gross pollutants are captured and retained efficiently while the stormwater exits through the retention screens and bypass channel. When the retention area is full, incoming gross pollutants escape via the bypass channel. This prevents congestion in the GPT inlet, overloading and upstream blockages in the stormwater conduits. Conversely, under adverse operating conditions involving low inlet flow rates and fully (100%) blocked screens, the gross pollutant capture and retention performance of the *LitterBank* GPT deteriorates rapidly [3].

To explore the adverse GPT operating characteristics, a 50% scale model experimental rig was placed in a tilting flume at the QUT hydraulic laboratory. The constant flow rate was established via controller settings on the centrifugal pumps which circulate the water from underground storage tanks into the flume. Flow rate readings were checked with periodical measurements at the flume outlet. Flow into the GPT was through a horizontal partially filled, 2 m rectangular section with an internal width of 146 mm. The height of the weir at the downstream end of the flume was fixed at 100 mm above the GPT floor. The measurements were obtained for a flow rate of 1.3 L/s through the GPT. Some small variations in the flow conditions ( $\pm 0.1$  L/s) during the course of the experiments were unavoidable, as a constant head tank was not fitted to the flume.

To study pollutant-free flow in a trap with fully blocked screens the GPT model was fitted with solid internal walls to represent 100% blockage. Neutrally buoyant particle seeding (20–50  $\mu\text{m}$ ) was introduced into the upstream GPT inlet flow via a feeding system. To obtain the flow structure within the GPT, the fluid motion of the particles was tracked with a high-speed camera (X-Stream<sup>TM</sup> XS-4) and by image acquisition (X-Vision version 1.13.05) software.

The results were processed using PIV software suite (proVision-XS version 3.08.30). The PIV system was supplied by Integrated Design Tools Inc. (IDT). Further details on the experimental setup are given in [1].

To investigate the gross pollutant capture and retention characteristics of a GPT, experiments were conducted with generic and custom modified large ( $\approx 40$  mm) celluloid spheres (table tennis balls). The variable density spheres were released simultaneously into the GPT inlet and their motions were recorded and analysed. Further details of the experimental setup are given by [3]. Experiments were repeated with sediments and their depositions recorded in various sections of the GPT.

### Texture-based flow visualisations

There are many different approaches to visualising flows including direct (point-based), geometric and texture-based methods. Texture-based flow visualisations are considered important as they provide a dense spatial coverage of the direction of the vector field and are able to handle dense vector datasets such as those collected by the PIV software. A comprehensive overview of texture-based methods is given by [4]. The two techniques used in this and previous work are based on an earlier approach, Line Integral Convolution or LIC [5, 6], and the more recent IBFV [7]. LIC is designed for steady flow visualisations and employs streamlines while IBFV employs pathlines and is able to visualise steady and unsteady flows (for steady flows, pathlines and streamlines are identical). Next we discuss the use of IBFV to visualise experimental GPT PIV datasets.

### Image based flow visualisation (IBFV)

IBFV was developed by [7]. It is based on advection and decay of textures in image space. Our method for generating IBFV animations of the GPT scattered vector field  $\vec{v}$  consists of two parts: an interpolation process followed by the application of IBFV. The interpolation process maps the scattered vector field  $\vec{v}$  onto a uniform, quadrilateral-based grid  $G_{xy}$  that is amenable to IBFV, although we note that IBFV can be applied to general polygonal shaped meshes (e.g., triangular). An overview of IBFV is given below followed by a more formal treatment.

The interpolation process uses SRFPACK, a fast, robust code for interpolating scattered data [8], to generate two-dimensional cubic spline interpolations of the irregularly spaced stream ( $U_x$ ) and crosswise ( $U_y$ ) velocities [9]. This results in two smooth surfaces,  $F_x$  and  $F_y$ , that interpolate  $U_x$  and  $U_y$  respectively. The interpolating surfaces  $F_x$  and  $F_y$  are used to map the stream and crosswise velocities on the regular uniform grid  $G_{xy}$ . We denote the interpolated vector field lying on  $G_{xy}$  as  $V = [V_x, V_y]^T$ .

IBFV is an iterative texture based method that generates animations of unsteady flow fields  $\vec{v}(\vec{x}, t)$ . At iteration  $k+1$ , an animation frame is generated which represents the flow at time  $t_{k+1}$ . Frame  $k+1$  is generated using the frame  $k$  blended with a high contrast, background noise image  $G_{k+1}$  (pixel values usually set to 0 or 1). The noise image  $G_{k+1}$  is selected in round-robin fashion from a set of noise images  $G_0, \dots, G_N$ . These are computed in advance by applying a periodic function  $f$  to the pixels of the initial noise image  $G$ . As a result, the images  $G_k$  are temporally coherent (that is they animate smoothly). The coherence is achieved by using  $f$  to smoothly oscillate pixel intensities. Advection and blending of the oscillating pixel intensities creates a dense set of short pathlines which create detailed images of  $\vec{v}(\vec{x}, t)$ . A pathline  $\vec{x}_{path}(t)$  of  $\vec{v}(\vec{x}, t)$  is given by:

$$\frac{d}{dt} \vec{x}_{path}(t) = \vec{v}(\vec{x}_{path}(t), t) \quad (1)$$

IBFV can visualise steady flows  $\vec{v}(\vec{x})$  by repeated application of same field  $\vec{v}(\vec{x})$ . The IBFV algorithm can be readily implemented on PCs with graphic processing units (GPUs), producing lively high frame-rate flow animations.

### The IBFV method

IBFV repeatedly advects image pixel intensities along the pathlines  $\vec{x}_{path}(t)$  of  $\vec{v}(\vec{x}, t)$ . In fluid dynamics terms, each pixel intensity is represented as time dependent scalar particle property  $I(\vec{x}, t)$  that is advected by  $\vec{v}(\vec{x}, t)$ . We set  $I(\vec{x}, t_0) = G$ , where  $G$  is a randomly generated, high-contrast noise image with pixels randomly set to 0 or 1. The evolution of  $I(\vec{x}, t)$  is governed by the well-known material derivative and, as a consequence,  $I$  remains constant along pathlines. IBFV uses a uniform forward advection scheme to calculate the evolution of  $I(\vec{x}, t)$ . Time is discretised in equal steps  $t_k = k\Delta t, k = 0, 1, \dots, L$  and  $\Delta t$  is the uniform time interval. An Eulerian first-order integration scheme is applied to (1) and, noting the constancy of  $I(\vec{x}, t)$  along the pathlines, we have

$$I(\vec{x} + \vec{v}(\vec{x}, t_k)\Delta t, t_{k+1}) = I(\vec{x}, t_k) \quad (2)$$

A first-order Eulerian scheme is sufficient for short pathlines (higher order schemes could be used). The IBFV core is based upon the advection of the textures  $I(\vec{x}, t)$  and equation (2) is employed. At time  $t_k$ , the texture  $I(\vec{x}, t_k)$  is mapped over mesh  $M$ . Initially,  $M$  corresponds to the regular uniform grid  $G_{xy}$ . Equation (2) is then applied to the vertices of  $M$ . This distorts the mesh  $M$  and the texture  $I(\vec{x}, t_k)$  by the flow field  $\vec{v}(\vec{x}, t_k)$ . An elegant aspect of IBFV is that advection and distortion of the texture  $I(\vec{x}, t_k)$  can be efficiently implemented on GPUs. The advection process is coded as a texture mapping operation and computation proceeds in a highly parallel fashion using the GPU hardware. The texture-mapped image of  $I(\vec{x}, t_k)$  is then set to  $I(\vec{x}, t_{k+1})$  and the process is repeated.  $I(\vec{x}, t_{k+1})$  is advected by  $\vec{v}(\vec{x}, t_{k+1})$  to produce  $I(\vec{x}, t_{k+2})$  and so on.

Repeated advection of the initial texture  $I(\vec{x}, t_0)$  causes difficulties without some intervention. This is most noticeable at the edges of the flow domain. At boundary inflows, the mesh will move away from the edge resulting in gaps within the textures  $I$ . IBFV overcomes this by including a noise injection term  $G(\vec{x}, t_k)$ . A scalar  $\alpha \in [0, 1]$  sets the blending ratio of advected noise  $I$  to injected noise  $G$  as follows:

$$I(\vec{x} + \vec{v}(\vec{x}, t_k)\Delta t, t_{k+1}) = (1 - \alpha)I(\vec{x}, t_k) + \alpha G(\vec{x} + \vec{v}(\vec{x}, t_k)\Delta t, t_{k+1}) \quad (3)$$

However, the injected noise  $G(\vec{x}, t_k)$  needs attention to ensure the IBFV frames still animate smoothly. If the noise term is randomly generated, random or ‘‘jerky’’ animations result. A time coherent noise term  $G(\vec{x}, t_k)$  is employed in (3). It is calculated using an initial random, high-contrast noise texture  $G(\vec{x})$  which is modulated over time by a smooth periodic function  $f$  to form  $G(\vec{x}, t)$ . The intensity of the injected noise at pixel  $\vec{x}$  oscillates smoothly between 0 and 1.  $G(\vec{x}, t)$  is defined as follows:

$$G(\vec{x}, t) = f\left(\left(t + G(\vec{x})\right) \bmod 1\right), \text{ where} \quad (4)$$

$f: \mathfrak{R}_+ \rightarrow [0, 1]$  is periodic with a period of 1

If  $\alpha = 1$  in (3), the animation consists solely of injected noise  $G(\vec{x}, t)$  and pixel intensities will vary continuously from black (0) to white (1) to black. If  $\alpha < 1$ , noise advection is introduced. [7] experimented with several functions  $f(t)$  and concluded that a square wave produced animations with superior contrast due to its sharp edged profile. We also preferred and used the square wave  $f(t) = 1$ , when  $t < \frac{1}{2}$  and 0 otherwise.

## Implementation of the IBFV method

Noise injection can be efficiently implemented using a set of pre-generated noise images  $G_n(\vec{x}, t), n = 0, 1, \dots, N$  that sample  $G(\vec{x}, t)$  at  $N$  equally spaced points over one period of  $f$ .

Typically  $16 \leq N \leq 64$ . We set  $N = 64$ . Since  $f$  is periodic, the noise term  $G$  can be selected in round-robin fashion from  $G_n(\vec{x}, t)$ .  $G$  is “injected” into the image using an alpha blending operation. The blending and advection operation can be implemented using the PC’s GPU, as described above. Hence, the calculation of (3) can be accelerated by the PC’s GPU, enabling the creation of lively fluid flow animations.

## IBFV and classic experimental fluid dynamics

Intuitively, IBFV animations can be compared to the classic dye injection technique used in experimental fluid dynamics. The continuous, periodic variation of a single pixel intensity of injected noise is representative, in experimental terms, of the injection of dye into the fluid flow at the location of the pixel. When  $0 < \alpha < 1$ , “dye is injected” and advected by  $\vec{v}(\vec{x}, t)$ . In terms of experimental fluid dynamics, IBFV models experiments where dye streams are injected at all pixels. This results in the generation of short pathlines from all pixels and an IBFV animation represents the evolution of this dense set of pathlines.

## Results and Discussion

The discussion begins with the average statistically processed PIV data obtained by experimentally capturing the seeded flow in the GPT with neutrally buoyant particles. It was previously shown that the vector plots from PIV image processing software in Figure 2 were shown to be visually cluttered due to the high resolution of the two-dimensional grid velocity dataset [1]. To overcome the visual clutter created by the standard vector visualisation methods employed by the PIV software, two texture based vector visualisation techniques—LIC and IBFV—were applied to the collected raw PIV vector data. Unlike conventional streamline plots, the LIC images were produced with a higher order interpolation scheme to avoid minor irregularities in some of the flow features that were observed when using the PIV visualisation software [1].

Some irregularities or distorted flow patterns are noted in the LIC images, for example, small dark patches behind the baffle and at the corner of the GPT. Dark patches in the main flow which cause obvious discontinuities, are experimental artefacts due to either a lack of seeding or to the fact that the overhead structures supporting the baffle and inner wall in the GPT obscure the camera sighting.

Despite the clarity of the LIC images, some important aspects of the visualisations are still unclear. For example, flow directions are not obvious particularly at the corners of the GPT which play an important role in the gross pollutant capture retention characteristics of the GPT. The velocities in the LIC images have also been normalised. Although, this technique is useful for highlighting the details of the lesser flow features, the visualisation of the high and low shear velocity gradients is not clear despite the colour mapping.

To overcome these limitations, a program has been developed in this paper to implement the IBFV method using the experimental dataset. It is based upon van Wijk’s sample code [10]. The program animates the flow in the GPT using the experimental dataset and several user display options are available. These options include displaying the velocity flow fields in un-normalised and normalised formats. Regions of high and low velocities are detected by the speed of their animation when using the un-normalised format (Figure 4). For example, in Figure 4, the higher velocity streamlines are more distinct as

shown by the blurred effect of their flow paths. These flow paths were found to be consistent with the CFD predicted flow features [2] and have been correspondingly labelled items 1-7 in Figure 4.

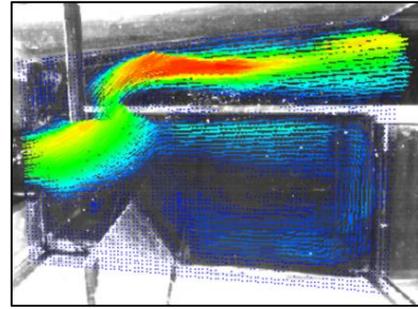


Figure 2. A typical vector plot produced by the PIV software of averaged velocity data.

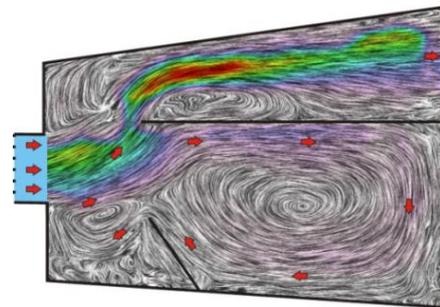


Figure 3. Application of the LIC algorithm to a typical flow field dataset.

The normalised IBFV animation (not shown) snapshot highlights well the lesser flow features and the experimental artefacts—as shown in black regions—in relation to the static LIC images in Figure 3. However, it is noted that the normalised format generated by the program does not represent the actual physics of the flow field. For example, the speed of the animation of certain flow features relative to the mainstream flow is not consistent with experimental observations. The un-normalised images in Figures 4 and 5 show clearly the flow features of zones 1-7. However, when animated the difficulty of determining the velocity direction in the static LIC images is completely resolved.

An additional program feature is the ability to visually trace and highlight streamlines using a tracer marker to investigate specific areas of the GPT as shown in Figure 5. Exchange of fluid between regions aids the understanding gross pollutant capture/retention characteristics. Hence, this was visualised using tracer markers as shown in Figure 5. For example, in the convoluted flow from zone 2—the diverticulum—and back into the retention area of the GPT is shown by the dark blue tracer in Figure 5. During the capture/retention experiments, generic spheres were seen to oscillate between these regions along this convoluted flow path [3].

The bulk of the fluid from the jet inlet to GPT outlet via the bypass channel is shown by the magenta tracer marker which has been fully discussed in [2, 11]. This behaviour implies that the majority of the incoming gross pollutants will escape the retention area via the bypass channel. This leads to the GPT’s poor capture and retention of gross pollutants [3]. It is also observed, that the remaining spheres in the retention area subsequently escaped via the bypass channel, further adding to the poor capture/retention performance of the GPT. Conversely, observations with the finer stormwater pollutants resulted in sediment depositions in the retention area.

The accumulation of waste in the corners of the GPT, that is the dead zones (See zones 3, 4, 7 in Figure 4), was also investigated with circular tracer markers.

It has been shown previously that dead zones play an important role in the stormwater pollutant capture/retention characteristics of a GPT [2]. Sediment depositions and to a lesser extent spheres, were particularly observed in zones 2 - 4 and 7 (Figure 4). In zone 4—at the top left corner of the GPT—a higher concentration of sediment was noted in comparison with the depositions in the regions behind the baffle, next to the inlet (zones 2 and 3). This behaviour can be attributed to the high and low velocities measured in the regions of the bypass channel and inlet/baffle, respectively [2, 11]. The green blob tracer marker (See 4, Figure 5) denotes that effluent in dead zones can have very long residence times.

A comparison of Figures 4 and 5 and the experimental observations show that the deposition of sediments and artificial gross pollutants is determined by at least four main factors: proximity to the wall, the size of the recirculation zones, the corresponding velocities and the interchange of fluid between the mainstream flow and the lesser flow features. This was indicated by the varying degrees of sediment and gross pollutant depositions observed in all flow feature zones (Figure 4).

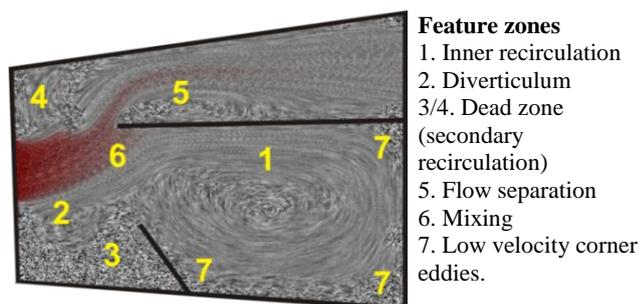


Figure 4. Snapshots of the IBFV with un-normalised velocity vectors.

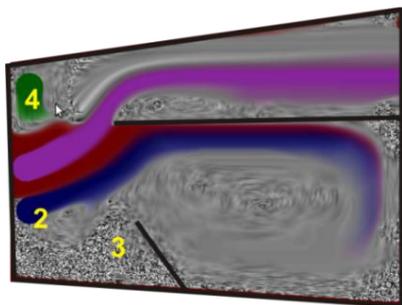


Figure 5. Snapshots of the IBFV with streamline tracers (for feature zones 2 - 4, see Figure 4). The tracers were introduced at the inlet (magenta) and dead zones at top (green) and bottom left (blue) corners of the GPT. The black outlines in the snapshots are the solid boundaries walls.

## Conclusions

A typical flow field in a stormwater GPT involving fully blocked screens and a low flow rate was experimentally captured and visualised. A technique was developed to apply the IBFV algorithm to the experimental raw dataset. The technique facilitated the characterisation of flow feature within the GPT.

The animation of flow and the interactive environment provided by the IBFV method resolved some of the flow ambiguities which had arisen from the static LIC images. These ambiguities related to the direction of flow and the movement/mixing behaviour of fluid in various parts of the GPT. Such hydrodynamic behaviour was further investigated by analysing results from deposition experiments with sediments and artificial gross pollutant particles.

The results indicated that the deposition of particles appears to rely on at least three main factors: proximity to the wall, the size

of the recirculation zones and the corresponding velocities of fluid within these zones.

It is concluded that the IBFV is a useful visualisation and analysis tool in investigating the hydrodynamic and stormwater pollutant capture/retention characteristics of a GPT. Furthermore, the IBFV provides an interactive environment to investigate flow fields that is ideally suited to teaching and learning purposes. Further work is underway to analyse a range of flow regimes using results obtained from experiments and CFD.

## Acknowledgments

The authors acknowledge C-M Concrete Pty. Ltd, 2004 (Mr Phil Thomas) as industry partner, an ARC linkage grant, the assistance of Ms Sarita Gupta Madhani and Mr David Warne.

## References

- [1] J. T. Madhani, J. Young, N. A. Kelson, and R. J. Brown, "A novel method to capture and analyze flow in a gross pollutant trap using image-based vector visualization," *Water, Air, & Soil Pollution: Focus*, vol. 9, pp. 357-369, 2009.
- [2] J. T. Madhani, N. A. Kelson, and R. J. Brown, "An experimental and theoretical investigation of flow in a gross pollutant trap," *Water Science and Technology*, vol. 59, pp. 1117-1127, 2009.
- [3] J. T. Madhani and R. J. Brown, "Experiments on particle capture and retention characteristics of a gross pollutant trap with highly blocked screens," *Journal of Hydraulic Engineering (Submitted)*, 2012.
- [4] R. Laramee, S., B. Jobard, and H. Hauser, "Image Space Based Visualization of Unsteady Flow on Surfaces," in *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, Seattle, Washington, USA, 2003.
- [5] B. Cabral and L. C. Leedom, "Imaging vector fields using line integral convolution," presented at the Proceedings of the 20th annual conference on Computer graphics and interactive techniques, Anaheim, California, USA, 1993.
- [6] D. Stalling and H.-C. Hege, "Fast and resolution independent line integral convolution," in *The 22nd Annual Conference on Computer Graphics and Interactive Techniques-SIGGRAPH 95*, Los Angeles, California, 1995, pp. 249-256.
- [7] J. J. van Wijk, "Image based flow visualization," in *Proceedings of the 29th annual conference on computer graphics and interactive techniques*, San Antonio, Texas, 2002, pp. 745-754.
- [8] R. J. Renka, "Algorithm 752: SRFPACK: software for scattered data fitting with a constrained surface under tension," *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, pp. 9-17, 1996.
- [9] R. J. Renka and A. K. Cline, "Scattered data fitting using a constrained Delaunay triangulation," *International Association for Mathematics and Computers in Simulation (IMACS) Transactions on Scientific Computing: AI, Expert Systems, and Symbolic Computation*, vol. 3, pp. 208-214, 1992.
- [10] J. J. van Wijk. (2002, 23-26 July). Image based flow visualization: supplemental material. *Sample code*. Available: <http://www.win.tue.nl/~vanwijk/ibfv/>
- [11] J. T. Madhani, N. A. Kelson, J. Young, and R. J. Brown, "Recirculating flow characteristics in a gross pollutant trap with fully blocked screens.," *Journal of Hydraulic Engineering (Submitted)*, 2012.