

Reconstructing the Geometry of Flowing Water

Ivo Ihrke Bastian Goldluecke Marcus Magnor
MPI Informatik, Saarbruecken, Germany
{ihrke, bg, magnor}@mpi.de

Abstract

We present a recording scheme, image formation model and reconstruction method that enables image-based modeling of flowing bodies of water from multi-video input data. The recorded water is dyed with a fluorescent chemical to measure the thickness of a column of water, which leads to an image formation model based on integrated emissivities along a viewing ray. This model allows for a photo-consistency based error measure for a weighted minimal surface, which is recovered using a PDE obtained from the Euler-Lagrangian formulation of the problem. The resulting equation is solved using the level set method.

1. Introduction

Recently, new multi-view reconstruction problems, different from the traditional diffuse surface reconstruction, have emerged in the field of computer vision. These include multi-view reconstruction of time-varying, transparent, natural phenomena like fire and smoke [6, 5, 1].

The work so far concentrates on non-refracting media. In this paper, we present a level set method for the reconstruction of a time-varying free flowing water surface. This problem arises in the context of free-viewpoint video, where we are concerned with the automatic acquisition of dynamic models for computer graphics purposes. The main problem here is that the surface structure can not be determined with traditional methods due to refraction effects, implying a complex image formation process. We alleviate this problem by dyeing the water with a fluorescent chemical. This allows us to directly measure the thickness of the water volume as a ray passes through it and hits the CCD-chip of the camera. In addition, a sophisticated energy minimization method is utilized for the reconstruction process, which is able to correctly incorporate error functions depending on surface normals. Obviously, this is a vital requirement if one wants to take into account refraction.

Image-based modeling of natural phenomena suitable for free-viewpoint video is performed using sparse view tomographic methods [6, 1] or surface based methods [5].

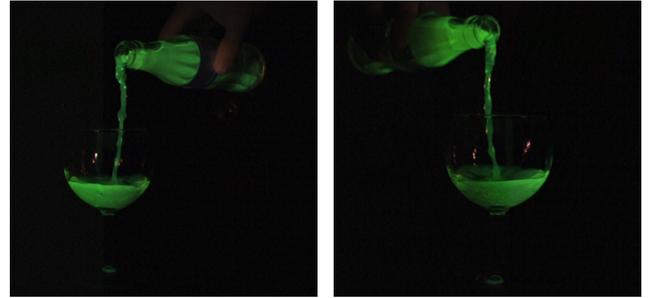


Figure 1. Source images from two of the cameras for one frame of our test video sequence, in which we pour fluorescent water from a bottle into a glass.

Reche et al. reconstruct trees from still images [11]. In [9], the geometry of hair is retrieved using a single camera and varying light source positions, exploiting the anisotropic reflectance properties of hair.

Only limited work has been done which directly addresses image-based reconstruction of water. In [8], a time-varying water surface is obtained by analyzing the distortion of a known texture beneath the water surface using optical flow and shape from shading techniques. Schultz [12] studies the reconstruction of specular surfaces using multiple cameras. However, both of these methods can only determine a height field for a rectangular surface area, while we reconstruct fully three-dimensional bodies of water.

Another line of research is refractive index tomography e.g. [10, 14]. These methods usually need expensive apparatuses and do not lend themselves to image-based modelling. Whereas refractive index tomography attempts to reconstruct a field of *varying* refractive indices, we know that we have a constant refractive index and need to compute the surface of a volumetric body of water.

Our paper is organized as follows. Sect. 2 defines the reconstruction problem we want to deal with and presents a mathematical justification for the level set surface flow yielding an optimal solution. Details for the implementation using PDEs are discussed in Sect. 3. We present results obtained with both synthetic 2D data as well as recorded 3D data of flowing water in Sect. 4, and conclude with ideas for future work in Sect. 5.

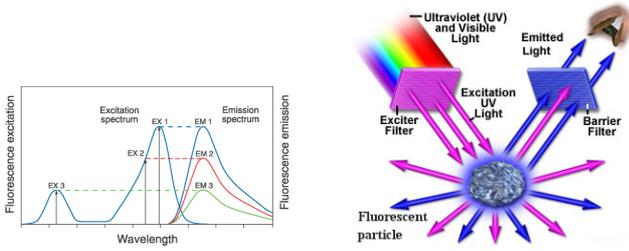


Figure 2. Left: Excitation and emission in fluorophores: the excitation wavelength changes the amplitude of the emission spectrum only, the profile stays the same. Right: The use of filters generates a proper excitation light source, and allows the observer to measure the emitted spectrum without interference from the excitation light source.

2. General Reconstruction Problem

Our goal is to reconstruct the surface area of a possibly moving body of water, using recordings from only a handful of fully calibrated cameras distributed around the scene. In order to be able to work with a well-defined image formation model, special care has to be taken when acquiring the water video data. We employ a fluorescent dye which causes the water to emit visible light when exposed to UV radiation. An example input image from a single frame is shown in Fig. 1.

This section embeds the reconstruction problem we want to deal with in a rigorous mathematical framework. Subsection 2.1 discusses the image formation model underlying the optimization. It shows how to generate synthetic views given a certain reconstructed surface Σ , which can be compared to recorded real-world data in order to define a photo-consistency error measure. The 'best' surface is determined by minimizing an error functional optimizing photo-consistency. The functional is defined in subsection 2.2, while the mathematical foundations for its minimization using a level set surface flow are addressed in subsection 2.3. After the theoretical discussion in this section, we proceed with the details of the implementation in Sect. 3.

2.1. Image Formation Model

We dissolve the chemical Fluorescein in the water. Fluorescein exhibits a photo-luminescent behavior i.e. it has the ability to absorb light of higher energy and subsequently re-radiate light with a lower frequency than the light used for excitation. Fig. 2 explains this principle. The emission spectrum is independent of the excitation wavelength, only the amplitude of the emitted light changes. A schematic of our studio setup is shown on the right hand side. We place filters in front of the light source and the cameras, respec-

tively. The two filters allow us to measure the emitted light only, which in turn lets us treat the body of water as a self-emissive medium.

We evenly dissolve the dye in the water and use a strong UV source to illuminate it. This allows us to assume a constant fluorescent emissivity throughout the volume. Thus, the accumulated light intensity along a ray traced through the water can be computed by multiplying its total length within the volume with a constant emittance ρ . Furthermore, a color calibration on the cameras is performed, such that they exhibit a linear response to the incoming light intensity, scaling light intensity to image intensity by a factor of γ .

Now, let p be a point in the image plane of camera C , and c be the camera's center of projection. We want to compute the theoretical pixel intensity $I_{\Sigma}(p)$ in the presence of a surface Σ , enclosing a volume V_{Σ} of water prepared as above. Let $R(c, p)$ be the ray traced from c in the direction of p through the surface Σ , taking into account correct refraction, Fig. 4. We ignore scattering and extinction effects in the water volume. Then,

$$I_{\Sigma}(p) = \gamma \int_{R(c,p) \cap V_{\Sigma}} \rho ds = \rho\gamma \int_{R(c,p) \cap V_{\Sigma}} ds.$$

The last integral just measures the length the ray traverses through V_{Σ} . In order to avoid having to determine the constant factor $\rho\gamma$ experimentally by acquiring and measuring a calibration scene, we implement an auto-calibration scheme. All image intensities are divided by the average intensity of the pixels in the image within the silhouette, and all ray-traced intensities by the average intensity of the rays corresponding to these pixels. The resulting quotients are independent of the quantity $\rho\gamma$.

Now that we are able to compute synthetic views given a surface Σ , we have to determine how well a reconstructed surface fits a given set of input views. If we are able to quantify the error, it can be used to define an energy functional mapping surfaces to real numbers, whose minimum yields an optimal reconstruction result. This aim is pursued in the next subsection.

2.2. Energy Minimization Formulation

We have to take care of photo-consistency of a reconstructed surface Σ with the given source images. We set up an energy functional

$$\mathcal{A}(\Sigma) := \int_{\Sigma} \Phi(s, \mathbf{n}(s)) dA(s), \quad (1)$$

defined as an integral of the scalar valued weight function Φ over the whole surface. $\Phi(s, \mathbf{n})$ measures the photo-consistency error density, and may depend on the surface

point s and the normal \mathbf{n} at this point. The larger the values of Φ , the higher the photo-consistency error, so the surface which matches the given input data best is a minimum of this energy functional. Because refraction occurs frequently, the dependency of the error measure on the normal is a vital part of our method, in contrast to many other previous applications of weighted minimal surfaces in computer vision.

The question remains how to correctly choose the error measure. Ideally, we would want it to be the difference of the measured intensity in every camera with the theoretical intensity, which would look something like this:

$$\Phi_{\text{naive}}(s, \mathbf{n}) := \sum_{i=1}^n (I_{\Sigma, i}(s) - I_i \circ \pi_i(s))^2,$$

where $I_{\Sigma, i}(s)$ is the ray-traced image intensity assuming surface Σ , I_i is the i th image, and π_i the i th camera's projection mapping.

While the general idea is good and exactly what we implement, in this initial form it faces several problems, the worst of which is that we have to be able to evaluate the error function away from the surface in order to perform the surface evolution later. We postpone the exact technical definition to Sect. 3, in favor of a discussion of the general mathematical tools with which to find a minimum of an energy functional of the form above.

2.3. Level Set Surface Flow

Instead of implementing a surface evolution directly, we will make use of the level set idea. We express the surfaces Σ_τ for each parameter value $\tau \geq 0$ as the zero level sets of a regular function

$$u : \mathbb{R}^3 \times \mathbb{R}^{\geq 0} \rightarrow \mathbb{R}, \quad u(s, \tau) = 0 \Leftrightarrow s \in \Sigma_\tau. \quad (2)$$

We require $u(\cdot, \tau)$ to be negative inside the volume enclosed by Σ_τ , and positive on the outside.

As we proved in [4], we arrive at a local minimum of the error functional if we choose a good initial starting surface σ_0 and evolve this surface according to the evolution equation

$$\frac{\partial}{\partial \tau} u = \left[-\operatorname{div} \left(\Phi \cdot \frac{\nabla u}{|\nabla u|} \right) + \operatorname{div}_\Sigma(\Phi \mathbf{n}) \right] |\nabla u|, \quad (3)$$

which we have to implement.

3. Implementation

In this section, we go into the details on how to implement our reconstruction scheme. Subsection 3.1 specifies

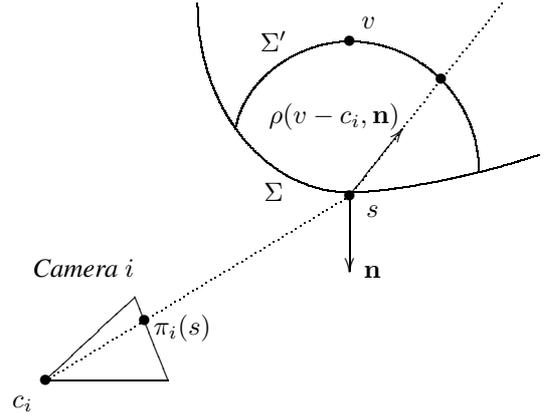


Figure 3. Evaluation of the partial error function ϕ_i for a single camera.. The length difference between rays traced through the distorted surface Σ' and the undistorted surface Σ is just $\|s - v\|$. Note that \mathbf{n} is not necessarily the exact surface normal, it may vary close to it in order to evaluate the derivative of Φ with respect to the normal.

the construction of the error function. For a stable evolution, we have to make sure that the surface does not shrink below the image silhouettes. We finally describe the implementation of the PDE as a narrow band level set method in subsection 3.2.

3.1. Construction of the Error Function

Of particular difficulty is the evaluation of the error function $\Phi(s, \mathbf{n})$ for a given point s and corresponding normal \mathbf{n} . The problem is that this term has to be evaluated away from the current surface Σ in order to compute the derivatives in (3), i.e. for points that do not lie directly on the surface, and with a normal which may be different from the current surface normal. The particular question one asks in that case is what local error would arise if the surface was distorted such that it lies in s with normal \mathbf{n} . For this reason, ray tracing in order to evaluate the error function has to be performed for a distorted surface Σ' . The computation of $\Phi(s, \mathbf{n})$ is thus performed in three steps.

In the first step, we construct the distorted surface Σ' through which rays are traced. We have to change Σ locally in a reasonably smooth manner such that the new surface passes through s . At this moment, we do not yet care about the normal. Assume for now that s lies outside the volume V_Σ enclosed by Σ . The desired result can then be achieved by uniting V_Σ with a ball B centered in the point v closest to s on Σ , with radius $\|s - v\|$. Vice versa, if s lies inside V_Σ , then we can achieve the result by subtracting B from V_Σ , Fig. 3.

The second step is to define the set of cameras $\mathcal{C} = \{C_1, \dots, C_k\}$ which contribute to the error measure. Ide-

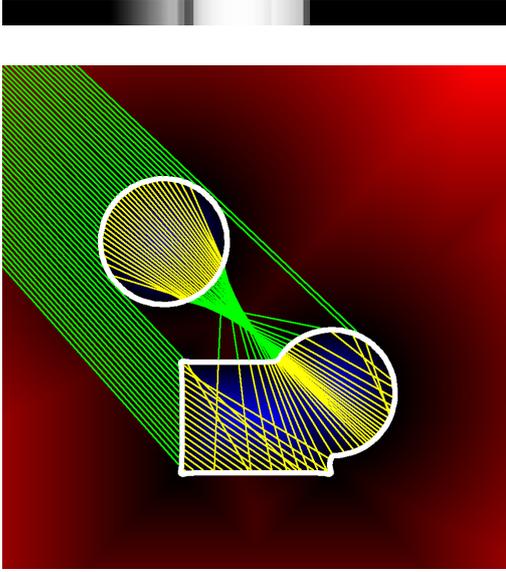


Figure 4. The rays used to generate a view from the upper left direction, visualizing the complexity of the image formation process. Top: resulting 1D view, intensity of each pixel is proportional to the length of the yellow segments for the corresponding ray.

ally, since the medium is transparent, we would like to consider all cameras we have available. Unfortunately, this would require to find for each camera the ray passing from the camera center to s , possibly refracted multiple times on the way. This computation definitely is too time-consuming. Instead, we only consider those cameras which have a good enough unobscured view of v with regard to the original surface. More precisely, each camera C_i belonging to \mathcal{C} must meet the following two criteria:

- The straight line from v to the center of projection c_i must not intersect Σ , and
- The ray starting from v in the refracted direction $\rho(v - c_i, \mathbf{n})$ must travel inside V_Σ in the beginning. ρ is computed using Snell's law, using the index of refraction of water for inside the volume, and of vacuum for outside.

In the third step, we finally compute the photo-consistency error ϕ_i for each contributing camera C_i and average those to get the total error Φ . Each individual error is computed as follows: Let $I_i \circ \pi_i(s)$ be the intensity of the projection of s in image I_i , and $r_i(s, \mathbf{n})$ be the accumulated intensity along a ray traced from s into the refracted direction $\rho(s - c_i, \mathbf{n})$. Then

$$\phi_i(s, \mathbf{n}) := (I_i \circ \pi_i(s) - r_i(s, \mathbf{n}))^2.$$

This corresponds to comparing the image intensity to the

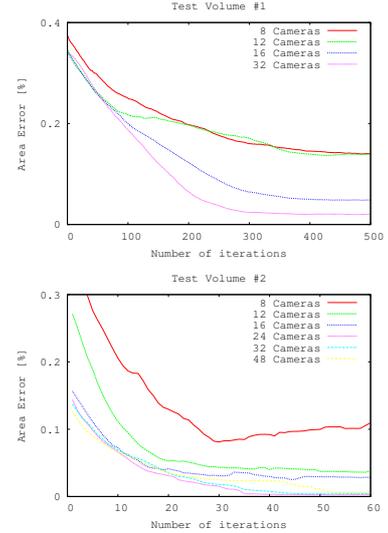


Figure 5. Convergence of the results depending on the number of input views.

ray-traced intensity of a ray cast from the camera to s , refracted as if by a surface located in s with normal \mathbf{n} . Thus, the desired normal \mathbf{n} is also correctly taken into account.

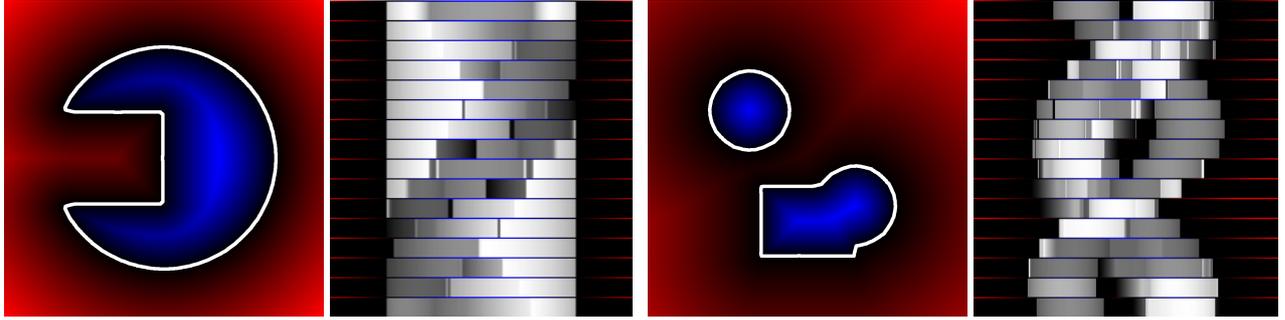
3.2. PDE Discretization

In order to implement the level set evolution equation, the volume surrounding the surface Σ has to be discretized. We use a regular three-dimensional grid of evenly distributed cells with variable spatial resolution of usually 64^3 or 128^3 cells. The surface is evolved according to the narrow band level set method [13], starting the evolution with the visual hull surface Σ_0 and the values u_0^{xyz} of the corresponding level set function u_0 in the centers of the grid cells. The values of the level set function are updated iteratively using the upwind scheme. At iteration step $i + 1$, the new values u_{i+1}^{xyz} are obtained from the values u_i^{xyz} of the previous iteration step by a discrete version of equation (3) using an explicit time step. To ensure stability, the step size $\Delta\tau$ must be chosen such that the level sets of u_i cannot cross more than one cell at a time, i.e. satisfy the CFL-condition.

4. Results

4.1. Synthetic 2D Experiments

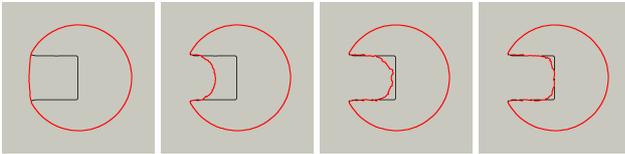
In order to verify that our surface evolution is capable of producing correct results despite the complex problem we want to solve, we first test it on synthetic 2D data. We ray-trace several views of two different test volumes using the image formation model presented in Sect. 2.1. The first volume is designed to test how well the algorithm can recover



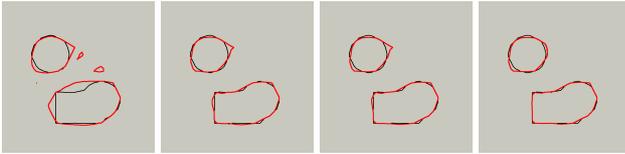
(a) The first synthetic volume together with 16 input views. Below each view is shown the signed distance transform σ of the silhouette.

(b) The second synthetic volume, also together with 16 input views and signed distance transform of the silhouette.

Figure 6. Synthetic test volumes and ray-traced views. Red color denotes positive values of signed distance, blue color negative values.



(a) Convergence towards the first test volume, after 0, 100, 200, and 300 iterations.



(b) Convergence towards the second test volume, after 0, 15, 30, and 45 iterations.

Figure 7. The best results we achieved using 24 input views, together with several in-between stages of the iteration.

concavities, while the second volume is not connected and has a mixture of straight and round edges. Both test volumes and resulting 1D views are shown in Fig. 6. An exemplary trace through the volume can be found in Fig. 4. This trace gives a glimpse of the complexity of the reconstruction problem, and demonstrates how heavily the ray-tracing result depends on the normals.

We run our algorithm with different numbers of input views in order to test the dependence of convergence on this critical parameter. The results are shown in Fig. 5. Convergence becomes stable with eight or more cameras used, with twelve views required in the more complex second test case. We can also note that there is a quick saturation of reconstruction quality with respect to the number of cameras. The visual hull does not improve much more if more than

16 cameras are used, in accordance with earlier results [7]. In addition, the quality of the reconstruction peaks at around 24 cameras for both test volumes. Interestingly, more cameras do not necessarily imply a better result, which indicates that a good placement of the cameras is at least as important as their sheer number. The best reconstruction results were achieved with the moderate number of 24 cameras, shown in Fig. 7.

In all cases, the algorithm runs with the same parameter values of $\epsilon_1 = 0.1$ and $\epsilon_2 = 100$. It exhibits a very stable behaviour against parameter changes, as the following table suggests. Here, 24 Cameras are used for the estimation of the first test volume, and the error after exactly 200 iterations depending on different parameter values is noted down.

		ϵ_1				
		0.01	0.1	0.5	1	5
ϵ_2	1	0.07	U	U	U	U
	10	0.05	0.04	0.06	U	U
	50	0.16	0.07	0.03	0.04	U
	100	0.04	0.05	0.04	0.06	U
	1000	S	S	S	S	0.03

As a rule of thumb, there is a certain threshold value for the speedup term above which it accelerates the evolution above a stable limit, causing the surface to shrink uncontrolled below the silhouettes. This is indicated by a “U” in the table. Too low a choice of ϵ_1 has no ill effects on stability, but slows down the convergence a bit. ϵ_2 can safely be chosen somewhere between 10 and 100 without much effect, but may cause the surface to be stuck at an undesirable spot if set too high, as indicated by the “S” in the table.

4.2. Real-world Water Videos

For the real-world tests, we use a multi-video studio consisting of 8 CCD-cameras with a resolution of 1004×1004 pixels. The cameras can record at a frame-rate of 45 frames

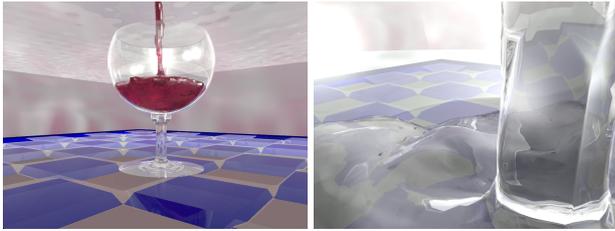


Figure 8. Reconstructed stream of water placed in a virtual environment. Left: Turning water into wine - we changed the material properties of the water such that it resembles red wine. Right: Close-up of the water surface, showing the details of the reconstructed geometry.

per second. A 300W UV light source is employed to illuminate the Fluorescein-dyed water. We acquire test sequences using a dark studio, the excitation light source and the fluorescent water being the only source of light. This measure allows for simple background subtraction. The reconstruction is performed on an equidistant, uniform grid of 128^3 voxels. An example of a reconstructed water surface rendered in a virtual environment and with changed material properties is shown in Fig. 8.

5. Summary and Conclusions

We have presented a method for the reconstruction of flowing water surfaces suitable for free-viewpoint video. A novel recording methodology and a corresponding image formation model allow us to define a photo-consistency constraint on the reconstructed surface. We utilize weighted minimal surfaces to refine the visual hull of the water using constraints based on thickness measurements of the real surface. The resulting energy functional is minimized using the Euler-Lagrange formulation of the problem, leading to a partial differential equation. This PDE is solved by applying the well known level set method. Synthetic tests indicate that the solution of the equation is stable. Real-world tests demonstrate the suitability of our method for the reconstruction of water.

Our Future work includes research into the applicability of our method to the reconstruction of other refractive media. Additionally, we would like to develop a hierarchical representation of the underlying computational grid to achieve a higher resolution reconstruction which allows to resolve finer details.

References

- [1] Lukas Ahrenberg, Ivo Ihrke, and Marcus Magnor. Volumetric Reconstruction, Compression and Rendering of Natural Phenomena from Multi-Video Data. *to appear*, page to appear, June 2005.

- [2] V. Caselles, R. Kimmel, G. Sapiro, and C. Sbert. Three-dimensional object modeling via minimal surfaces. In *Proc. European Conference on Computer Vision*, volume 1, pages 97–106. Springer, April 1996.
- [3] O. Faugeras and R. Keriven. Variational principles, surface evolution, PDE's, level set methods and the stereo problem. *IEEE Transactions on Image Processing*, 3(7):336–344, March 1998.
- [4] Bastian Goldluecke and Marcus Magnor. Weighted minimal hypersurfaces and their applications in computer vision. In *European Conference on Computer Vision (2)*, volume 3022 of *Lecture Notes in Computer Science*, pages 366–378, Prague, Czech Republic, May 2004. Springer.
- [5] S. W. Hasinoff and K. N. Kutulakos. Photo-Consistent 3D Fire by Flame-Sheet Decomposition. In *In Proc. 9th IEEE International Conference on Computer Vision (ICCV '03)*, pages 1184 – 1191, 2003.
- [6] Ivo Ihrke and Marcus Magnor. Image-Based Tomographic Reconstruction of Flames. *ACM Siggraph / Eurographics Symposium Proceedings, Symposium on Computer Animation*, pages 367–375, June 2004.
- [7] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *Proceedings of ACM SIGGRAPH*, pages 369–374, 2000.
- [8] H. Murase. Surface Shape Reconstruction of a Nonrigid Transparent Object Using Refraction and Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(10):1045–1052, October 1992.
- [9] S. Paris, H. M. Briceño, and F. X. Sillion. Capture of Hair Geometry from Multiple Images. *ACM Transactions on Graphics*, 23(3):712–719, August 2004.
- [10] C. Pintavirooj, A. Romputtal, A. Ngamlamiad, W. Withayachumnankul, and K. Hamamoto. Ultrasonic Refractive Index Tomography. *Journal of WSCG*, 12(2):333–339, February 2004.
- [11] A. Reche, I. Martin, and G. Drettakis. Volumetric Reconstruction and Interactive Rendering of Trees from Photographs. *ACM Transactions on Graphics*, 23(3):720–727, August 2004.
- [12] H. Schultz. Retrieving Shape Information from Multiple Images of a Specular Surface. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):195–201, February 1994.
- [13] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Monographs on Applied and Computational Mathematics. Cambridge University Press, 2nd edition, 1999.
- [14] Andrei V. Zvyagin, K. K. M. B. Dilusha Silva, Sergey A. Alexandrov, Timothy R. Hillman, and Julian J. Armstrong. Refractive index tomography of turbid media by bifocal optical coherence refractometry. *Optics Express*, 11(25):3503–3517, December 2003.