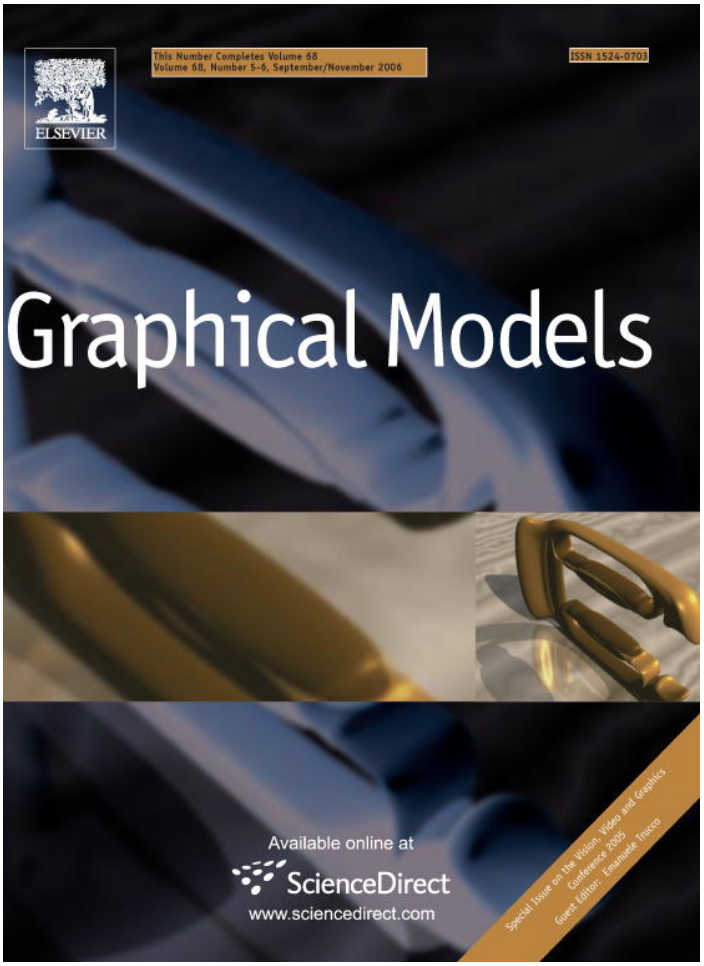


Available online at www.sciencedirect.com
ScienceDirect
Graphical Models 68 (2006) 484–495
Graphical Models
www.elsevier.com/locate/gmod

Adaptive grid optical tomography

Ivo Ihrke [a,*], Marcus Magnor [b,1]

[a] Graphics—Optics—Vision, MPI für Informatik, Stuhlsatzenhausweg 85, 66 123 Saarbrücken, Germany
[b] Institut für Computergraphik, TU Braunschweig, Mühlenpfordtstraße 23, 38 106 Braunschweig, Germany

Received 31 January 2006; received in revised form 3 June 2006; accepted 27 August 2006

Abstract

Image-based modeling of semi-transparent, dynamic phenomena is a challenging task. We present an optical tomography method that uses an adaptive grid for the reconstruction of a three-dimensional density function from its projections. The proposed method is applied to reconstruct thin smoke and flames volumetrically from synchronized multi-video recordings. Our adaptive reconstruction algorithm computes a time-varying volumetric model, that enables the photorealistical rendering of the recorded phenomena from arbitrary viewpoints. In contrast to previous approaches we sample the underlying unknown, three-dimensional density function adaptively which enables us to achieve a higher effective resolution of the reconstructed models.

Keywords: Image based reconstruction; Dynamic volumetric models; Natural phenomena

1. Introduction

Dynamic natural phenomena like fire and smoke are challenging to model realistically [4,17,22]. We propose to use real images of such phenomena to obtain computer models that are suitable for photo-realistic image synthesis. In [11] optical tomography is introduced as a suitable method to reconstruct volumetric phenomena from camera images. The method is applied to reconstruct three-dimensional volumetric models of flames. This article presents an advancement of this scheme that is more memory-efficient which allows increasing the resolution of the reconstruction. Besides flames, we reconstruct another class of volumetric phenomena, i.e. thin smoke.

Image-based modeling of transparent phenomena has received only little attention in computer vision. There have been approaches to extend surface reconstruction by taking transparency into account [3,21]. Computerized tomographic (CT) methods have been applied to rigid body reconstructions [5]. Transparent, volumetric phenomena are treated by Hasinoff et al. [8,7]. In [8] the *flame sheet decomposition* algorithm is developed, which reconstructs a surface (the flame sheet) with varying transparency and color. Ihrke and Magnor [11] use 3D-CT reconstruction to generate time-varying volumetric models of flames (See Fig. 1).

In contrast, this paper presents an adaptive grid computerized tomography technique that has the

* Corresponding author. Fax: +49 681 9325 599.
E-mail addresses: ihrke@mpi-sb.mpg.de (I. Ihrke), magnor@cg.tu-bs.de (M. Magnor).
[1] Fax: +49 531 391 2103 (M. Magnor).

1524-0703/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.gmod.2006.08.001

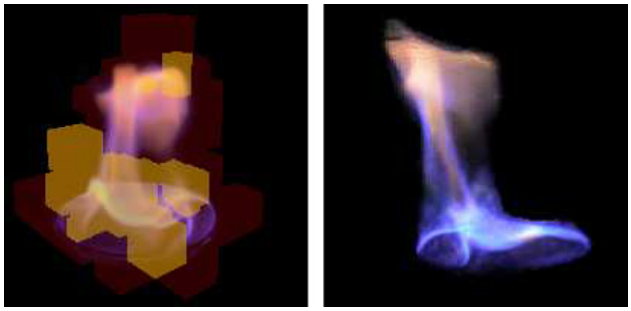


Fig. 1. Left: an intermediate step of the adaptive reconstruction process, the yellow boxes indicate regions of large error. Right: the reconstruction result. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this paper.)

advantage of requiring less memory and allowing for higher resolution reconstructions. It uses an octree data structure [20] to manage the inhomogenous set of basis functions. The closest work to this are multigrid CT reconstruction [14,9] and wavelet-based multi-resolution tomography [2,18,19]. The latter works use multi-resolution techniques to restrict the number of X-ray exposures [18], to suppress noise in smooth regions of the reconstructed image [19] and to establish an error bound on the reconstructed image [2]. In the CT literature the term adaptive computerized tomography is applied to methods that perform the data acquisition process adaptively e.g. [10]. Octree-based adaptive methods have also been successfully used in fluid dynamic simulations to increase the effective resolution of the simulation [15].

The paper is organized as follows: Section 2 reviews the uniform grid algorithm and its underlying assumptions. A discussion of the applicability to the reconstruction of smoke follows. In Section 3 we derive the adaptive algorithm. Section 4 presents experiments and results. In Section 5 we discuss properties and sources of error of our method before we conclude the paper and present directions for future work in Section 6.

2. Review of the basic algorithm

In this section, we review the computerized tomography (CT) algorithm presented in [11]. In this paper, a basic computerized tomography technique is described, that is based on an algebraic formulation of the inversion of a simplified image formation model for fire. We discuss its applicability to smoke in Section 2.4.

2.1. Image formation model

Hasinoff and Kutulakos [8] present a simplified image formation model for fire. The fire is modeled as a 3D density field ϕ of fire reaction products, i.e. soot particles. Image intensity is related to the density of luminous particles in the fire. The model has the form

$$I_p = \int_c \phi \mathrm{d}s + I_{\mathrm{bg}}. \tag{1}$$

Here, I_p is pixel p's intensity, c a curve through 3D space, ϕ is the density field of soot particles and I_{bg} is the background intensity. Curve c is the backprojected ray of pixel p. We approximate every pixel by one ray through the density field. The underlying assumptions of this simplified model are

– Negligible absorption/scattering—this assumption is valid for fire not substantially obscured by smoke, and
– Proportional self-emission—the brightness depends on the density of the soot particles only.

2.2. Mathematical derivation

In order to invert (1) we have to make an assumption on the structure of ϕ. We do this by assuming that ϕ can be represented as a linear combination of basis functions ϕ_i

$$I_p = \int_c \left(\sum_i a_i \phi_i \right) \mathrm{d}s + I_{\mathrm{bg}}. \tag{2}$$

The sum and the coefficients a_i are moved out of the integral and we get

$$I_p = \sum_i a_i \left(\int_c \phi_i \mathrm{d}s \right) + I_{\mathrm{bg}}. \tag{3}$$

Eq. (3) describes a linear system of equations,

$$\mathbf{p} = \mathbf{S}\mathbf{a} + \mathbf{b}. \tag{4}$$

The rows represent the equations for one pixel and the columns contain the integrals of the pixel's backprojected rays over the basis function ϕ_i. See Eq. (9) and the following discussion for details. The choice of the basis functions ϕ_i is essential for the tractability of the problem. The box basis function

$$\phi_i^{\mathrm{Box}}(x,y,z) = \begin{cases} & x^i_{\min} < x \leqslant x^i_{\max} \\ 1 & y^i_{\min} < y \leqslant y^i_{\max} \\ & z^i_{\min} < z \leqslant z^i_{\max} \\ 0 & \text{else} \end{cases} \tag{5}$$

is a popular choice because of its simplicity. This is especially true in an adaptive setting because basis functions with shared support would require the introduction of asymmetric basis functions at subdivision level boundaries. Furthermore, the box basis function is non-negative which ensures a non-negative density field if a non-negative solution to Eq. (4) is found. This is necessary to ensure a physically plausible reconstruction.

2.3. Implementation issues

The system of Eq. (4) is generated in a similar way to volume raytracing [16]. We generate the complete system of equations, that means we incorporate one equation for every pixel in all camera images that contains all basis functions.

For a particular frame of a multi-video sequence we remove columns of the matrix in Eq. (4) corresponding to basis functions with non-zero support completely outside the visual hull [13] of the phenomenon. This effectively sets their coefficients to zero and makes the sparse view reconstruction process possible. This process is described in detail in Section 3.3. The resulting smaller linear system is solved in a least squares fashion

$$\mathbf{a} = (\mathbf{S}^{\mathrm{T}}\mathbf{S})^{-1}\mathbf{S}^{\mathrm{T}}(\mathbf{p} - \mathbf{b}). \tag{6}$$

The linear system, Eq. (6), is large and cannot be inverted directly. Furthermore, it is ill-conditioned, requiring some form of regularization to obtain a satisfactory solution. In general a trade-off between the smoothness of the solution and the accuracy of the data fit has to be found.

We compute the inversion using the CGLS variant [6] of the conjugate gradient (CG) method [1], which is an iterative solution method for linear systems. This variant of the CG method was developed for solving the normal equations found in least squares problems without explicitly computing $\mathbf{S}^{\mathrm{T}}\mathbf{S}$. This saves memory because the explicit representation of the matrix product is usually dense. The sparseness of $\mathbf{S}$ is very important for the computational efficiency of the inversion via iterative methods.

To obtain a non-negative solution a projection to the subspace of non-negative solutions is carried out in every iteration. This is done by setting all negative entries of the solution vector $\mathbf{a}$ to zero.

Hansen [6] discusses the regularizing properties of the CG method in detail. Despite the incomplete theoretical understanding of the convergence properties it is experimentally and partially theoretically shown that the CG method behaves quite similar to the truncated singular value decomposition. The singular values are captured in their natural order starting with the largest. This ensures that the main features of the solution are reconstructed first while components due to small eigenvalues are captured last. The components due to small eigenvalues are most severely affected by noise and amplified in the matrix inversion. Therefore, regularization methods aim at reducing the effects of these components on the solution.

As a termination criterion for the conjugate gradient iterations we adopt the L-curve criterion [1,6]. The quotient $\frac{||\mathbf{a}_k||_2}{||\mathbf{S}\mathbf{a}_k - \mathbf{p}||_2}$ of the norm of the solution at iteration k over the norm of the residual is analyzed. The point of highest curvature on this curve is the best trade-off between a smooth solution and accuracy in the fit [6]. The number of iterations of the CG method plays the role of the regularization parameter.

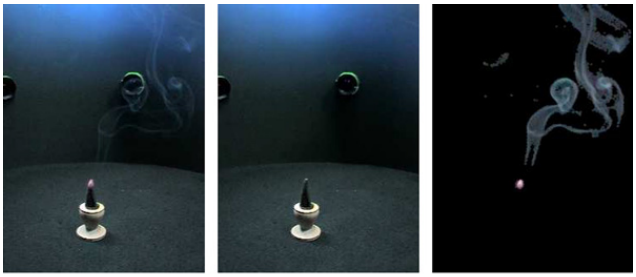
2.4. Application to smoke

To apply the presented method to smoke we have to make sure the image formation model is more or less valid. Both assumptions stated in Section 2.1 are obviously not true for smoke in general. We tackle the problem by making the following assumptions:

– the smoke is uniformly and diffusely lit, and
– scattering takes place in a uniform manner.

These assumptions make it possible to treat the smoke as a self-emissive medium. We found this model to be applicable for thin smoke reconstruction.

In the case of flame reconstruction it is possible to record in a dark setting, avoiding the complication of background subtraction for transparent phenomena, i.e. vector $\mathbf{b}$ of Eq. (4) is zero. Since the smoke has to be lit uniformly we have to perform background subtraction which involves some image processing. We record a sequence of background images and compute the median background image as well as the standard deviation for every pixel.



(a) Original (b) Background (c) Foreground

Fig. 2. Background subtraction for thin smoke: the original image (a) contains a burning incense that produces thin smoke. It is diffusely lit with daylight, (b) is a background image recorded beforehand and (c) is the background-subtracted image that contains only the smoke column.

This allows us to classify pixels into foreground and background in the smoke sequences. For pixels classified as foreground we subtract the median background value which corresponds to the additive nature of the image formation model. All other pixels are set to zero. This step computes the difference $(\mathbf{p} - \mathbf{b})$ in advance and is depicted in Fig. 2. In the following, we regard vector $\mathbf{p}$ as the background-subtracted pixel vector.

Reconstruction and re-rendering of the reconstructed model into the original views can be seen as a 2D image filtering operation [11]. This filter is defined by the image formation model and the reconstructed density field. Pre-processing of the input images might not correspond to a valid filter of this form and thus affect the reconstruction process negatively. Fortunately, background subtraction and noise reduction are not observed to have adverse effects on the visual quality of the reconstruction.

3. Adaptive reconstruction

An adaptive reconstruction technique for three-dimensional computerized tomography is motivated by the better memory efficiency that can be achieved. This in turn yields higher effective resolutions for the reconstructed models.

Using a uniform grid, the memory limit of 2 GB on a 32-bit machine is reached relatively fast. In our experiments, we have found that using eight cameras with images taken at 320×240 pixel resolution, we can achieve a reconstruction resolution of 128^3 voxels.

In general we have n_p rows with $O(\sqrt[3]{n_\mathrm{b}})$ non-zero elements each in matrix $\mathbf{S}$, where n_b is the number of basis functions needed to approximate the density field and n_p the number of pixels in a frame (a frame is one time frame of a multi-video sequence and contains n_c images, where n_c is the number of cameras). A uniform discretization of the reconstruction space is assumed. We use an index-stored sparse matrix as a representation for $\mathbf{S}$. We store the two indices and the matrix value for each non-zero entry. Even though this is the most memory-efficient storage scheme for an unstructured sparse matrix, the available memory fills up quite fast when using a uniform grid.

3.1. Basic iteration

An adaptive reconstruction algorithm has to proceed iteratively.

1. Estimate the coefficients $\mathbf{a}$ of Eq. (4), then
2. Project the residual error, i.e. image plane error onto the basis functions to get a measure where to split, followed by
3. Splitting of the k basis functions that are responsible for the largest error, and
4. Lifting of the current solution to the new discretization.
5. Until convergence go to step 1.

The following subsections cover the single steps in detail. The estimation process (step 1) has already been described in Section 2.2. It is independent of the shape of the basis functions and therefore directly applicable to the adaptive algorithm.

3.2. Error projection

The main difficulty in the adaptive estimation process is to relate the residual error

$$\mathbf{r} = \mathbf{p} - \mathbf{S}\mathbf{a} \tag{7}$$

to the interpolation error

$$|u - P_{\phi_i} u| \tag{8}$$

Here, u is the perfectly reconstructed function and $P_{\phi_i} u$ its projection onto the subspace of functions representable by the basis functions ϕ_i. A relation between the two errors allows for the identification of the coefficients $\mathbf{a_i}$ that contribute most to the residual error. We present a heuristic for this projection step and show the feasibility of an adaptive computerized tomography reconstruction.

The basic idea of our heuristic is based on the projection of the basis function's regions of support into the camera images, and on the accumulation of the residual error of the affected pixels, see Fig. 3. This yields an intuitive way of relating the error caused by a particular basis function to the residual error in the image plane. In addition the error measure is efficiently implemented using sparse matrix–vector multiplications.

Our main observation is that the complete geometry of the problem is encoded in the matrix **S**. The system of equations (4) has the following structure:

$$\begin{pmatrix} \mathbf{p_1} \\ \mathbf{p_2} \\ \vdots \\ \mathbf{p_{n_p}} \end{pmatrix} = \begin{pmatrix} \int_{c_1} \phi_1 \mathrm{d}s & \dots & \int_{c_1} \phi_{n_b} \mathrm{d}s \\ \int_{c_2} \phi_1 \mathrm{d}s & \dots & \int_{c_2} \phi_{n_b} \mathrm{d}s \\ \vdots & \vdots & \vdots \\ \int_{c_{n_p}} \phi_1 \mathrm{d}s & \dots & \int_{c_{n_p}} \phi_{n_b} \mathrm{d}s \end{pmatrix} \mathbf{a}. \tag{9}$$

Since our basis functions ϕ_i have local support the matrix **S** is sparsely populated. Note that every column of the matrix corresponds to a particular basis function. The rows are the equations for one particular pixel. Therefore, an entry $\mathbf{S_{ij}}$ is non-zero only if the support of basis function ϕ_j projects onto pixel $\mathbf{p_i}$. We use this observation to formulate the projection of the basis functions into the images and the accumulation of residual errors per basis function in matrix notation

$$\mathbf{e}_{\phi_j} = \frac{1}{n_{\phi_j}} \sum_i \delta_{ij}^S |\mathbf{r_i}| \tag{10}$$

$$\delta_{ij}^S = \begin{cases} 1 & \mathbf{S_{ij}} \neq 0 \\ 0 & \text{else.} \end{cases} \tag{11}$$

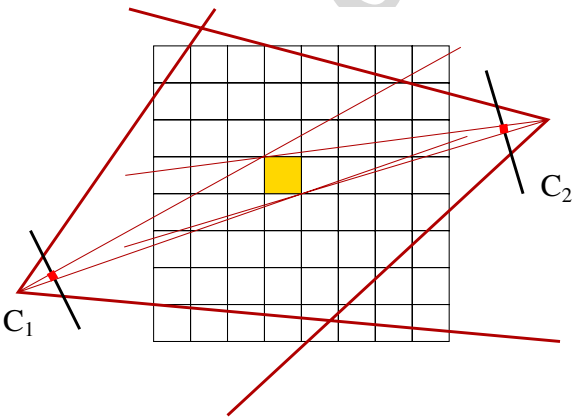


Fig. 3. The error heuristic measures the accumulated residual error in the image plane for each basis function.

The basis function ϕ_j is visible in n_{ϕ_j} cameras, and vector **e** contains the error measure for all basis functions. This computation requires either a specialized function in the implementation of the algorithm or a second copy of matrix **S** in main memory with all non-zero entries set to one. The specialized function approach might be expensive (e.g. index looping in Matlab). The copied matrix approach, on the other hand requires twice the amount of main memory. Therefore it is advisable, to find a better measure that does not require matrix **S** to be changed. By using the coefficients of matrix **S** directly we can incorporate a weight corresponding to the path-length of the backprojected ray c_i that is affected by basis function ϕ_j. This in itself is not sufficient to capture the influence of basis function ϕ_j on the error in pixel $\mathbf{p_i}$, because the coefficient $\mathbf{a_j}$ might scale basis function ϕ_j in an arbitrary way. A better version is therefore

$$\mathbf{e}_{\phi_j} = \frac{\mathbf{a_j}}{n_{\phi_j}} \sum_i \mathbf{S_{ij}} |\mathbf{r_i}|. \tag{12}$$

3.3. Splitting and basis function independent visual hull restriction

Using the error heuristic from the previous section, we iteratively split those k basis functions that cause the largest residual error. k is an arbitrary number that influences the convergence of the adaptive scheme. It should not be set too large because our error measure is just a heuristic and unnecessary basis functions might be introduced. We perform a uniform splitting on the box basis functions. For different types of basis functions the splitting process becomes more complicated. For example for the linear basis function, asymmetric basis functions have to be introduced where basis functions of different splitting level overlap.

We do not have to recompute matrix **S** in every iteration. Instead the columns corresponding to the split basis functions are removed and replaced by columns corresponding to the new basis functions, see Fig. 4. This, together with the error heuristic expressed in matrix form results in an efficient implementation of the iterative method described in Section 3.1.

The visual hull restriction of matrix **S** can be performed in an efficient and accurate way. The following discussion refers to Fig. 5. In step 1 we extract all rows that have non-zero entries in pixel vector **p**. These represent the rays that are inside

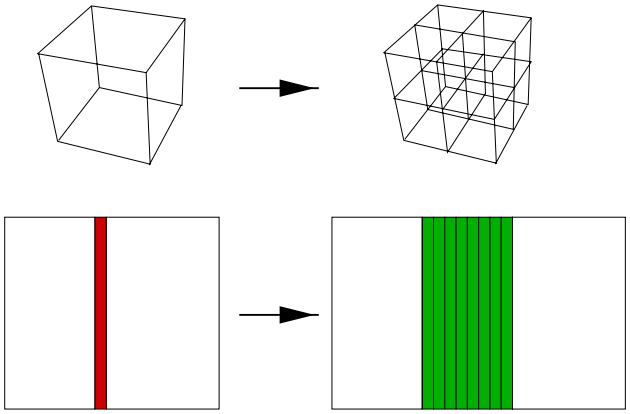


Fig. 4. Augmentation of matrix **S** to accommodate adaptive splitting.

the silhouette of camera c. In step 2 we identify the columns that have zero entries only, i.e. the basis functions that do not affect the silhouette in camera c. Therefore, they cannot be part of the visual hull. This step has to be performed per camera. Therefore, it is necessary to keep track where the pixels in vector **p** originated (see Fig. 6).

We compute a binary vector for each camera, marking all basis functions that are potentially contained in the visual hull with one. These correspond to columns containing non-zeros in the sub-matrices extracted in step 1. The basis functions marked with one have non-zero support in the generalized cone backprojected from the silhouette of camera c.

By taking the element-wise logical AND of all binary vectors, we compute the intersection of the

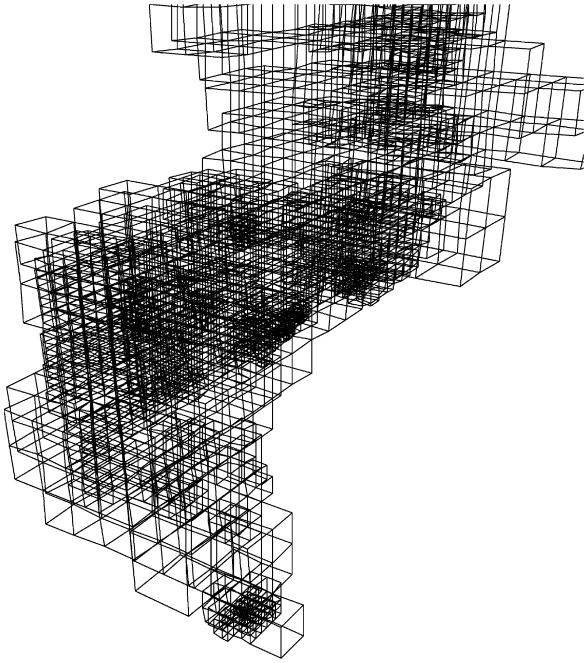


Fig. 6. A visual hull restricted adaptive grid (intermediate step in the iteration).

generalized cones of all cameras and thus the visual hull (step 3). This computation is accurate up to the discretization in the image plane, i.e. up to the pixel level. Step 4 restricts the original matrix **S** to columns corresponding to basis functions that have non-zero support in the visual hull. The resulting linear system has zero rows for some of the pixels in vector **p** that are outside the silhouette. Note that not all rows that have a zero right hand side get

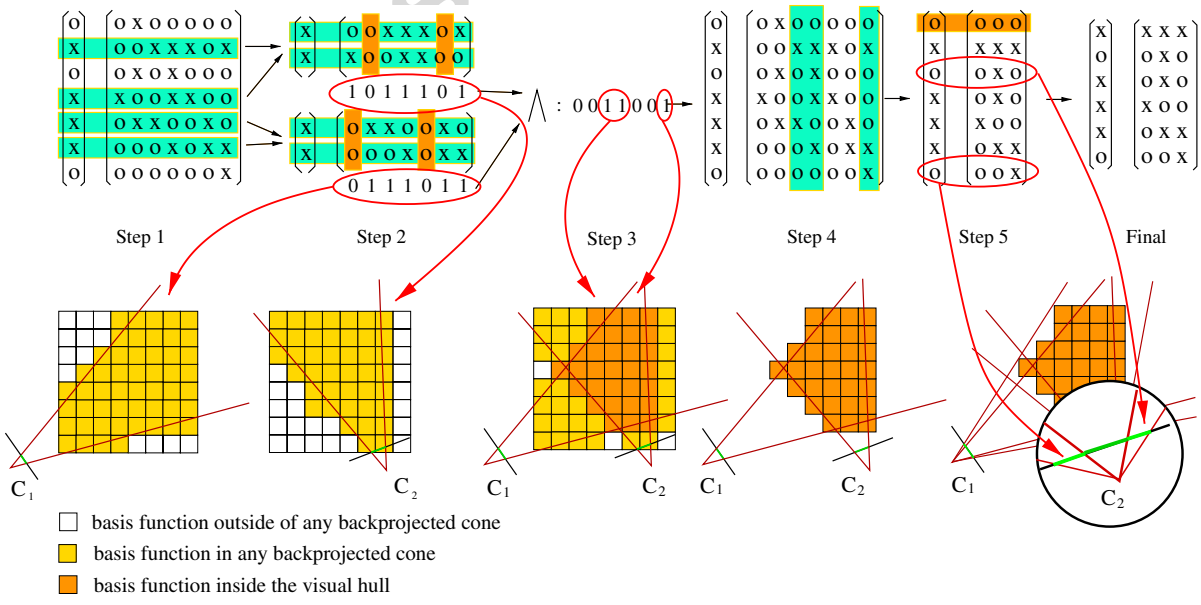


Fig. 5. Visual hull restriction of matrix **S**.

removed. This is because the basis functions might not be completely contained in the visual hull. Therefore, the zero pixels have to be accurately included in the estimation process (step 5). Basis functions on the boundary of the visual hull are very likely to be split, so the boundary of the visual hull gets represented accurately after some iterations of the adaptive scheme. This can be seen in the middle plot of Fig. 9.

The splitting process cannot be performed infinitely. It is advisable to set a maximum split level. A suitable criterion is the Nyquist limit, i.e. if a basis function projects to less than two pixels in all images the splitting can be stopped. A useful number for k is the square root of the number of basis functions currently used. This choice results in a sub-exponential growth in the number of basis functions but converges faster than a constant number k.

3.4. Initialization of the new discretization

The solution of the previous iteration in the adaptive process can be used as an initial guess for the updated discretization of the domain. This ensures a faster convergence of the conjugate gradient method in each iteration of the adaptive process. We initialize the coefficients of the unsplit basis functions with their value from the previous iteration and the newly introduced ones with the value of their parent in the octree data structure.

3.5. Implementation

The whole adaptive process can be efficiently implemented using basic matrix–vector operations. A simple indexed, unordered sparse matrix representation has been found to be suitable for the purposes of this algorithm. This allows for a straight-forward implementation and is also easily parallelizable. The memory requirements are typically only 20–25% of the uniform grid case while achieving comparable reconstruction accuracy. This allows for higher resolution input images and a higher resolution of the reconstructed model.

We use a minimalist octree data structure to keep track of the splitting process. For each column index of matrix $\mathbf{S}$ we store the split level from the root of the tree and the index that the corresponding leaf would have in a uniformly split octree under a fixed order of traversal. This enables us to identify the positions and sizes of the newly inserted basis functions and to generate the integral values for the new columns of matrix $\mathbf{S}$ in every iteration.

To minimize the overhead introduced by re-generating the matrix values for the newly inserted columns we keep track of the cameras and pixels that are affected by a particular basis function. For each row of matrix $\mathbf{S}$ we store the camera and the pixel corresponding to the linear equation. Thus by identifying the non-zero entries of the removed column we can efficiently generate the new non-zero entries of the columns replacing the original one. This is possible because the refined basis functions' support is completely contained in the support of the original basis function.

4. Results

4.1. Synthetic tests

To validate the proposed adaptive tomographic reconstruction scheme we performed tests on synthetic data. We use a phantom head model, Fig. 7, as is common in the computerized tomography literature [12] to assess the accuracy of tomographic reconstruction methods. The model includes stylized, typical features of a human head.

We use the head model as ground truth and compute the RMS error for different numbers of simulated cameras. We analyze the convergence properties of the adaptive scheme with respect to the number of basis functions used in the reconstruction process, Fig. 8. The plot shows the

Fig. 7. The Shepp-Logan phantom head model used as ground truth in our synthetic tests.

behaviour of the RMS error over 80 iterations of the adaptive scheme for different numbers of cameras. As expected the RMS error reduces with larger numbers of iterations and it converges on a lower level when using more cameras. The visual quality of the reconstructions is shown in Fig. 10.

4.2. Real-world experiments

For our real-world experiments we use a calibrated multi-camera setup with eight cameras. We record multi-video sequences at a resolution of 320×240 pixels with 15 frames per second. In the case of smoke we perform background subtraction as a pre-processing step. Because the background subtraction is not perfect, we use an alpha-matte in step 1 of the visual hull restriction process Fig. 5. The matte is created using morphological operators on the thresholded foreground images. The fire sequences are recorded in the dark and do not need to be pre-processed.

We performed a convergence study of the adaptive algorithm in a real-world setting. The results are shown in Fig. 9. Since we do not have ground truth data for this case we use the residual error for the analysis. The residual error decreases as expected with the number of iterations. The decrease is not monotonic though. This is because our error measure is based on a heuristic. An interesting graph is the plot of the number of rows in matrix $\mathbf{S}$ versus the number of iterations. It shows that the basis functions adapt to the pixel perfect visual hull. A visualization of the results after different numbers of iterations and the convergence of the solution is shown in Fig. 11. Along with Figs. 12 and 13 it shows reconstructions we have obtained by applying our algorithm to different multi-video sequences.

5. Discussion

Potential sources of error of our method are

– camera calibration errors,
– color calibration errors,
– 2D image processing, and
– discretization (number of views).

Geometric calibration errors are inevitable and tend to create too small visual hulls. This means that the actual silhouette is a bit larger than the silhouette of the reconstructed model. This effect is usually in the range of one or two pixels and can be considered negligible.

Our method relies on equal photometric properties of the imaging devices. Therefore, we perform a color calibration step using a Macbeth color checker on the acquired image sequences. Together with image processing operations prior to the model estimation a bias in the noise characteristics of the images is introduced. The effects of this bias have not been analyzed quantitatively but seem to be minor in

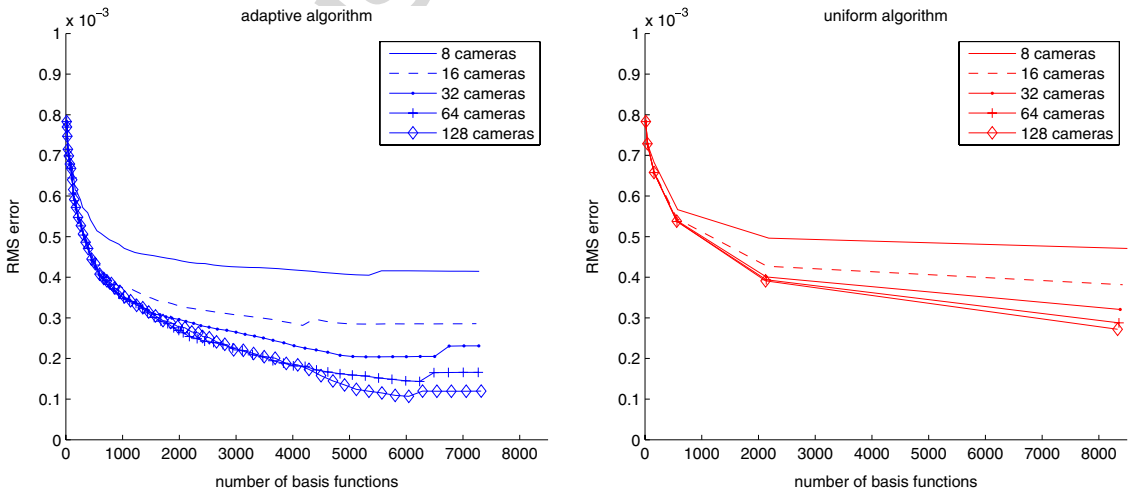


Fig. 8. RMS reconstruction error versus number of basis functions for different numbers of cameras. The plot on the left shows results for our adaptive algorithm, for comparison we ran the experiment with the same settings on a uniformly subdivided computational grid. The results are shown in the right plot. We note that the adaptive algorithm consistently achieves better results using less basis functions than the unfirom algorithm. The adaptive algorithm also outperforms the unfirom algorithm in overall reconstruction accuracy.

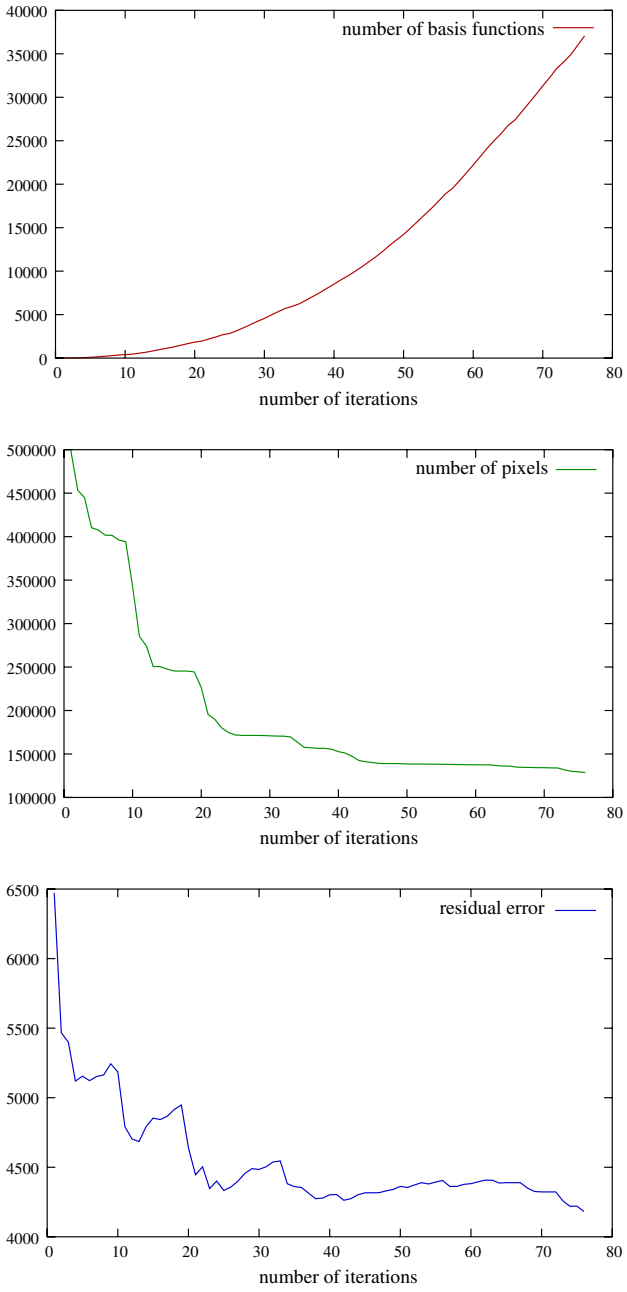


Fig. 9. Behavior of n_b the number of visual hull consistent basis functions (columns) in matrix $\mathbf{S}$ (top), n_p the number of pixels (rows) in matrix $\mathbf{S}$ (middle) and the residual error (bottom) versus the number of iterations of the adaptive algorithm run on real-world data.

comparison to the errors introduced by the sparse number of camera views.

While these error sources have an impact on the reconstructed density field, the presented method is sufficiently accurate for the modeling of three-dimensional, time-varying, volumetric phenomena for rendering purposes. We demonstrate that photo-realistic novel views can be synthesized from the recovered three-dimensional models.

Compared to the uniform discretization case our new method offers some advantages which are reduced memory requirements allowing for an increased resolution of the reconstruction. The adaptive grid on the other hand is a disadvantage for the reconstruction of video sequences in terms of computation time. Since the adaptive grid cannot be re-used for the different frames of the video sequence and the majority of the computation time is spent in the setup of the linear system, the reconstruction times for one frame with the adaptive method are comparable to the time spent for the reconstruction of a whole video sequence in the uniform grid case.

Thus the adaptive grid tomographic technique is primarily suited for the reconstruction of single frames where high resolution of the reconstructed model is of major concern.

6. Conclusions and future work

We have presented an adaptive algorithm for optical tomography. The algorithm is based on an octree hierarchy of piecewise constant basis functions. We propose a heuristic that enables the projection of errors in the image plane into the domain of the basis functions. This allows us to iteratively split basis functions that cause large residual errors in the image plane. Using this algorithm we are able to reconstruct dynamic, volumetric models of flames and thin smoke. Additionally we presented an efficient scheme for the accurate computation of the visual hull. This scheme is independent of the choice of basis functions and accurate up to the pixel level.

We believe that our adaptive tomography algorithm is applicable to other tomography problems as well. We would like to test it on real X-ray data to measure its performance compared to more traditional methods like filtered backprojection. Future work includes the derivation of a mathematically sound error projection as well as the use of different classes of basis functions. Wavelet bases provide an interesting option but it is more difficult to ensure a non-negative density field because the basis functions are not non-negative. This research would also provide a connection to wavelet-based multi-resolution schemes [2,18,19].

Further experiments regarding the combination of fire and smoke in recorded sequences should be conducted. Modeling occluding objects in flames and smoke is another possibility.

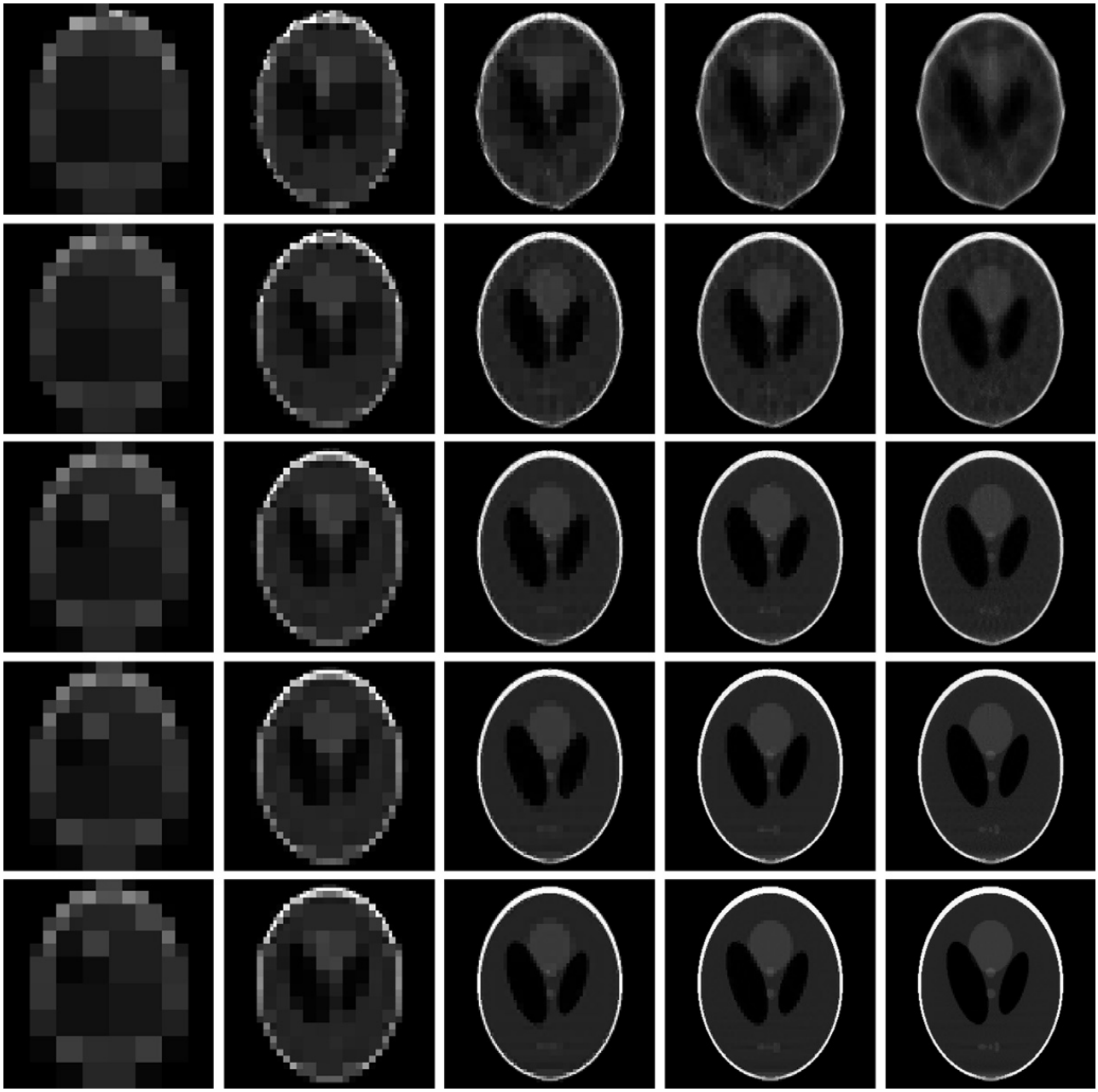


Fig. 10. Reconstruction results for different numbers of cameras (rows from top to bottom 8, 16, 32, 64, 128 cameras, respectively) and different numbers of iterations of the adaptive reconstruction scheme (from left to right 4, 12, 32, 48, 64 iterations).

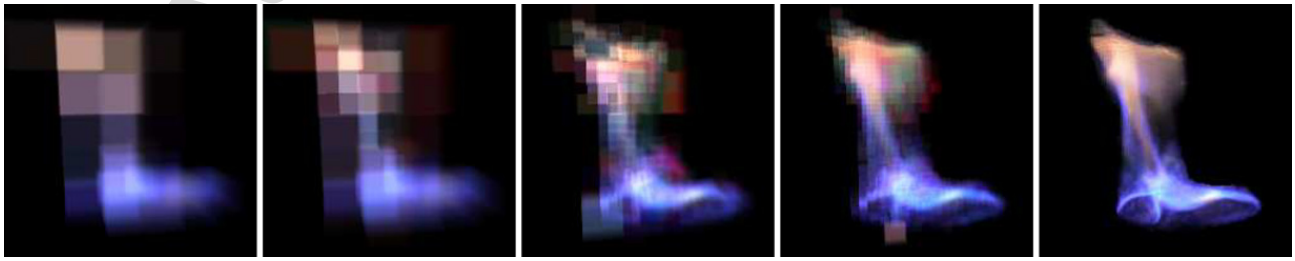


Fig. 11. Visualization of reconstruction results after 1, 2, 14, 28 and 100 iterations.

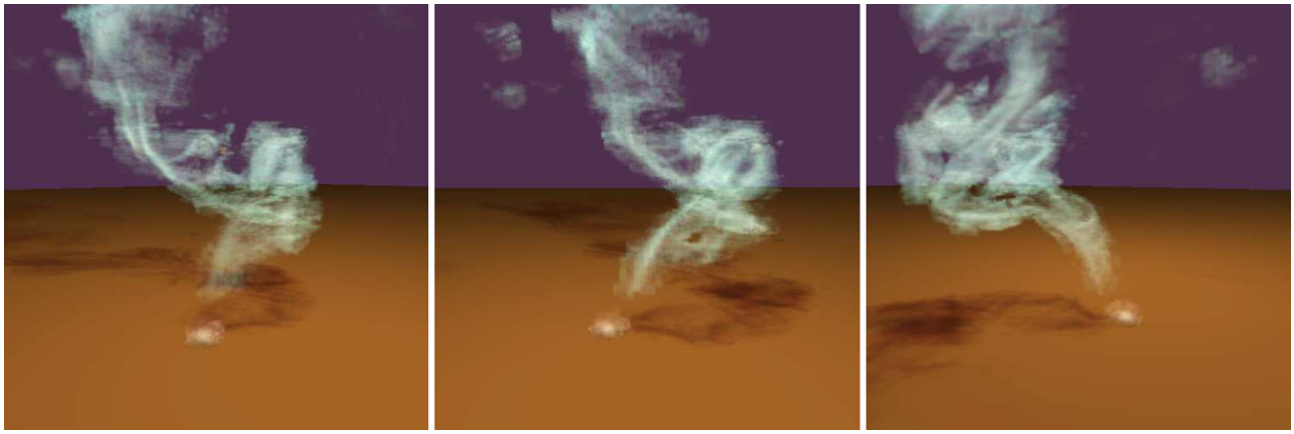


Fig. 12. A volumetric model of smoke rendered from different viewpoints.

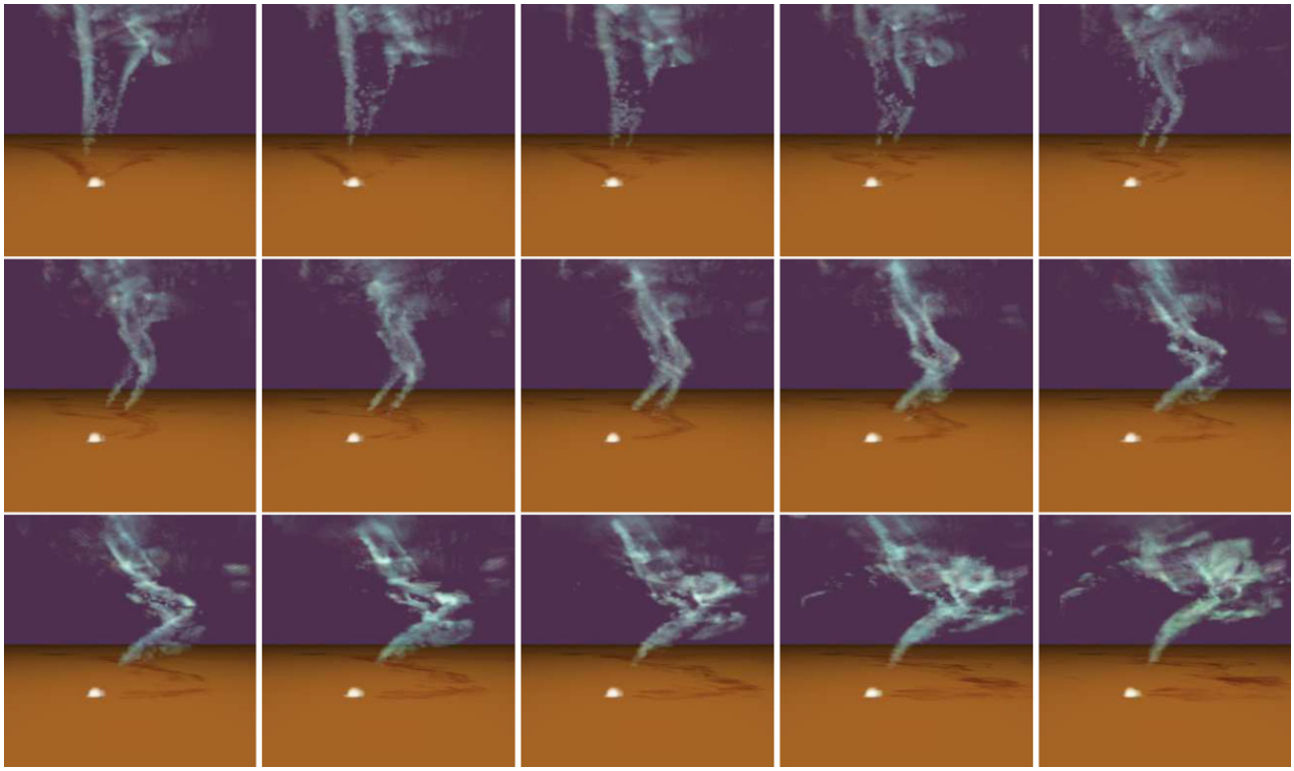


Fig. 13. Reconstructions of 15 consecutive frames of a smoke sequence.

Acknowledgment

This work is supported by the EC within FP6 under Grant 511568 with the acronym 3DTV.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.gmod.2006.08.001.

References

[1] Åke Björck, Numerical methods for least squares problems, Society of Industrial and Applied Mathematics (1996).
[2] M. Bathia, W.C. Karl, A.S. Willsky, A wavelet-based method for multiscale tomographic reconstruction, Transactions on Medical Imaging 15 (1) (1996) 92–101.
[3] J.S.D. Bonet, P.A. Viola, Roxels: responsibility weighted 3D volume reconstruction, in: Proceedings of the International Conference on Computer Vision (ICCV '99), 1999, pp. 418–425.
[4] R. Fedkiw, J. Stam, H.W. Jensen, Visual simulation of smoke, in: Proceedings of SIGGRAPH, 2001, pp. 15–22.

[5] D.T. Gering, W.M. Wells III, Object modeling using tomography and photography, in: Proceedings of IEEE Workshop on Multi-View Modeling and Analysis of Visual Scenes, 1999, pp. 11–18.
[6] P.C. Hansen, Rank-deficient and discrete Ill-posed problems, Society of Industrial and Applied Mathematics (1998).
[7] S.W. Hasinoff, Three-Dimensional Reconstruction of Fire from Images. MSc Thesis, University of Toronto, Department of Computer Science, 2002.
[8] S.W. Hasinoff, K.N. Kutulakos, Photo-consistent 3D fire by flame-sheet decomposition, in: Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV '03), 2003, pp. 1184–1191.
[9] V. Henson, M. Limber, S. McCormick, B. Robinson, Multilevel image reconstruction with natural pixels, SIAM Journal on Scientific Computing 17 (1996) 193–216.
[10] J. Hofmann, A. Flisch, A. Obrist, Adaptive CT scanning-mesh based optimisation methods for industrial X-ray computed tomography applications, NDT&E International 37 (2004) 271–278.
[11] I. Ihrke, M. Magnor, Image-Based Tomographic Reconstruction of Flames, in: ACM Siggraph/Eurographics Symposium Proceedings, Symposium on Computer Animation, 2004, pp. 367–375.
[12] A.C. Kak, M. Slanley, Principles of computerized tomographic imaging, Society of Industrial and Applied Mathematics (2001).
[13] A. Laurentini, The visual hull concept for silhouette-based image understanding, IEEE Transactions on Pattern Analysis and Machine Recognition 16 (2) (1994) 150–162.
[14] M.A. Limber, T.A. Manteuffel, S.F. McCormick, Optimal resolution in maximum entropy image reconstruction from projections with multigrid acceleration, in: Proceedings of the Sixth Annual Copper Mountain Conference on Multigrid Methods, 1993, pp. 361–375.
[15] F. Losasso, F. Gibou, R. Fedkiw, Simulation of water and smoke with an octree data structure, ACM Transactions on Graphics 23 (3) (2004) 457–462.
[16] N. Max, Optical models for direct volume rendering, IEEE Transactions on Visualization and Computer Graphics 1 (2) (1995) 99–108.
[17] D.Q. Nguyen, R. Fedkiw, H.W. Jensen, Physically based modelling and animation of fire, ACM Transactions on Graphics 21 (3) (2002) 721–728.
[18] F. Rashid-Farrokhi, K.J.R. Liu, C.A. Berenstein, D. Walnut, Wavelet-based multiresolution local tomography, Transactions on Image Processing 10 (6) (1997) 1412–1430.
[19] B. Sahiner, A.E. Yagle, Image reconstruction from projections under wavelet constraints, Transactions on Signal Processing 41 (12) (1993) 3579–3584.
[20] H. Samet, The Design and Implementation of Spatial Data Structures, Addison-Wesley Publishing Company Inc., 1990.
[21] R. Szeliski, P. Golland, Stereo matching with transparency and matting, in: Proceedings Sixth International Converence on Computer Vision, 1998, pp. 517–524.
[22] Y. Zhao, X. Wei, Z. Fan, A. Kaufman, H. Qin, Voxels on fire, in: Proceedings of the 14th IEEE Visualization Conference (VIS'03), 2003, pp. 271–278.