UCRL-TR-223676

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Analysis of Rayleigh-Taylor Instability Part I: Bubble and Spike Count

C. Kamath, A.Gezahegne, P. Miller

August 11, 2006

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

Abstract

The use of high-performance computers to simulate hydrodynamic instabilities has resulted in the generation of massive amounts of data. One aspect of the analysis of this data involves the identification and characterization of coherent structures known as “bubbles” and “spikes”. This can be a challenge as there is no precise definition of these structures, and the large size of the data, as well as its distributed nature, precludes any extensive experimentation with different definitions and analysis algorithms. In this report, we describe the use of image processing techniques to identify and count bubbles and spikes in the Rayleigh-Taylor instability, which occurs when an initially perturbed interface between a heavier fluid and a lighter fluid is allowed to grow under the influence of gravity. We analyze data from two simulations, one a large-eddy simulation with 30 terabytes of analysis data, and the other a direct numerical simulation with 80 terabytes of analysis data. We consider different techniques to first convert the three-dimensional data to two dimensions and then count the structures of interest in the two-dimensional data. Our analysis of the bubble and spike counts over time indicates that there are four distinct regimes in the process of the mixing of the two fluids, starting from the initial linear stage, followed by the non-linear stage with weak turbulence, the mixing transition stage, and the final stage of strong turbulence. We also show that our results are relatively insensitive to the parameters used in our algorithms.

Contents

1 Introduction

Hydrodynamic instabilities such as the Rayleigh-Taylor instability or the Richtmyer-Meshkov instability occur when one fluid is being accelerated by a second fluid. Such instabilities arise in diverse applications such as supernovae, oceans, and supersonic combustion, and are therefore the subject of much research. With the advances in high-performance computing, scientists are able to run high-fidelity, three-dimensional computer simulations of these instabilities. The resulting output is then analyzed to improve the understanding of these instabilities and construct models which can then be used in lower-fidelity simulations of physical phenomena.

These high-fidelity simulations enable scientists to capture the fine-scale detail of the instabilities as they evolve over time. However, they also produce massive amounts of data which require semi-automated techniques for their analyses. One aspect of the analysis involves the identification and characterization of coherent structures, referred to as “bubbles” and “spikes”, which develop as the two fluids mix. However, as these structures are not well defined, it can be a challenge to extract them from the data. This makes it difficult to obtain the statistics of interest such as the number or size of these structures, or understand their dynamical behavior.

From an analysis perspective, the problem is compounded by the fact that the data for these simulations is often stored in a distributed manner, with multiple files for each time step. This may require analysis algorithms which operate on the data piecemeal and then patch the results together. Alternatively, one could read the data back into a parallel machine for analysis. In addition, as the data is stored on longer-term, tape storage, accessing it can be time consuming. These data handling issues, combined with the size of the data and the lack of a precise definition for the structures of interest, can make the analysis of these datasets very challenging.

In this report, we focus on the problem of counting bubbles and spikes in simulations of Rayleigh-Taylor instability. We describe the problem and our analysis goals in Section 2, followed by a brief summary of related work in Section 3. We next describe the data from two 3-D simulations of Rayleigh-Taylor instability, one a large-eddy simulation (LES), and the other a direct numerical simulation (DNS). In Section 5, using 2-D slices from a small subset of the LES data, we illustrate the difficulties in counting the bubble and spike structures over the course of the simulations. We show that the structures are quite complex, especially at later time, and, in the absence of a clear definition of bubbles and spikes, it can be difficult to identify the extent of these structures prior to extracting the statistics of interest.

We next discuss our approach to identifying and characterizing the bubbles and spikes in the LES and DNS datasets. As these simulations were run on a regular Cartesian grid, we can treat the data as an “image” in 3-D and apply image processing techniques suitably extended to three dimensions and to process floating point values. In the first part of the analysis described in Section 6, we discuss how we can use image segmentation techniques in 3-D to first define the boundary of the bubbles and spikes and then convert the data into 2-D images. These 2-D images are then analyzed to count the bubble and spike structures as described in Section 7. We present our results for the two datasets in Section 8, followed by an analysis of the sensitivity of the results to the choice of various parameters in Section 9. We conclude with a summary and our plans for future work.

In this report, we focus on the bubble and spike counts for the LES and DNS datasets.

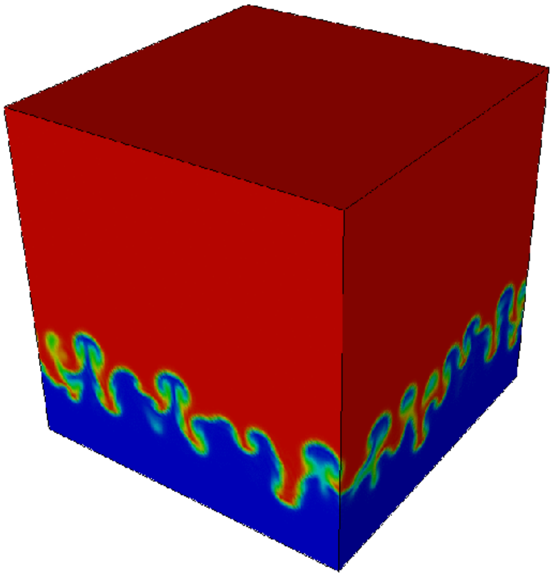


Figure 1: Bubbles and spikes in a 3-D LES simulation of the mixing of two fluids. The heavier fluid at top, in red, has density 3, while the lighter fluid at the bottom in blue has density 1. When an initially perturbed interface between the two fluids grows under the influence of gravity, fingers of the lighter fluid penetrate the heavier fluid as "bubbles" and "spikes" of heavier fluid enter the lighter fluid.

Other statistics of interest, such as distributions of bubble/spike sizes, distances between neighboring bubbles/spikes and the dynamics of their behavior will be addressed in a follow-on report.

2 Problem Description

We consider the Rayleigh-Taylor instability [14] which occurs when an initially perturbed interface between a heavier fluid which is on top of a lighter fluid is allowed to grow under the influence of gravity. The fingers of lighter fluid penetrate the heavier fluid in what are referred to as 'bubbles', while 'spikes' of heavier fluid move into the lighter fluid, as shown in Figure 1. With time, the bubbles and spikes, which are initially distinct, continue to evolve. In the process, they may grow, split, merge with surrounding bubbles (spikes), or shrink in size relative to other bubbles (spikes) which grow and overtake them.

In our study, we analyze two high-fidelity simulations of the Rayleigh-Taylor instability [5, 2], both generated using the Miranda code at Lawrence Livermore National Laboratory. Miranda is a research hydrodynamics code which can run in either a direct numerical simulation (DNS) or a large-eddy simulation (LES) mode. In the DNS mode, the viscosity and diffusion are directly simulated. In the LES mode, models are added to describe some of the finer-scale physical processes. As a result, a coarser grid can be used, reducing the computational time.

Both our datasets simulate the problem of Rayleigh-Taylor instability for a heavier fluid of density 3 on top of a lighter fluid of density 1, mixing under the influence of gravity. The three-dimensional computational domain is a cube with uniformly-spaced grid points, periodic boundary conditions in x and y, and no-slip walls imposed in z at the top and bottom of the cube.

Our first data set is from an LES simulation on an 1152^3 grid, with the initial interface perturbed as described in [5]. This simulation saved flow fields from 759 time steps. At each time step, seven variables are output in single precision at each grid point. These include the pressure, the density, the x, y, and z velocities, the viscosity, and the diffusivity. The total size of the analysis data is 30 terabytes.

Our second data set is from a DNS simulation under identical physical conditions to the LES simulation [2]. The grid is larger, at 3072^3, and the simulation saved flow fields from 248 time steps. Only five variables are output in single precision at each time step: the pressure, the density, and the x, y, and z velocities. The total size of the analysis data is 80 terabytes.

We note that the sample times at which the flow fields are saved for the LES and the DNS simulations are not physically or numerically equivalent.

2.1 Analysis Goals

The main goal of our analysis is to understand the evolution of the bubbles and spikes over time for both the LES and the DNS simulation data. We do this by extracting various characteristics of the bubbles and spikes, including their number, their size, the distances between them, and the number of neighbors. The dynamics of the bubbles and spikes are also of interest, in particular, whether they merge or split, whether the two (or more) bubbles/spikes which merge, grow as one or one grows at the expense of the other, etc. An analysis of how the bubble/spike statistics vary over time, along with their dynamics, provides insights into the physical processes in the mixing layer between the two fluids. In this report, we focus on the extraction of bubble and spike counts; other statistics will be addressed in a follow-on report.

In the rest of this report, when we discuss general techniques or make general observations, we will use the term ‘bubbles’ to mean both bubbles and spikes. In the results section, we will present results for each separately.

We encounter several challenges in this work. From an analysis viewpoint, a major problem is the lack of a precise definition of a bubble or spike, especially one which is consistent over time. At the early stages of the simulation, when the initial perturbation evolves to form distinct bubbles, we can easily identify each bubble visually. However, at later time, when the bubbles have started to merge, they are no longer distinct, and it is not possible to clearly identify the boundaries of all bubbles, even visually. A particular challenge is to define when, in the process of merging, can the two bubbles be considered as one. A similar issue arises when a single bubble splits into two or more. This identification of the bubbles is complicated by the fact that the surface of the bubbles is single valued only at the very early time steps.

Another challenge to the understanding of bubble behavior is related to the analysis algorithms, specifically, the choice of parameters for the analysis techniques and the sensitivity of the results to these parameters. As we discuss further in Section 5, there is a range of scales for the objects of interest. The bubbles start out as small perturbations, just a few grid points off the interface between the two fluids. By the end of the simulation, the mixing layer grows to occupy over 70% of the grid in the z direction. At this stage, small perturbations of a few grid points can be considered as noise. This range of scales, where the signal at the early time steps is similar to the noise at the later time steps, implies that a single analysis algorithm is unlikely to work for all time steps. For the same reason, a fixed set of parameters for all time steps may not be the optimal choice for extracting the necessary information. To ensure that our results are an accurate reflection of the data, rather than the analysis algorithms and their parameters, we conduct sensitivity analysis on the choice of parameters and compare the results obtained using different algorithms. We also prefer analysis algorithms which depend on few parameters, are relatively easy to implement, and are appropriate for the way in which the data at any time step are distributed among multiple files.

The sheer size of the data is also a major challenge to the analysis, especially when coupled with the imprecise definition of the bubbles and the algorithm issues with extracting the statistics of interest. The latter imply that we need to experiment with different ways of defining and extracting the bubbles, especially at the later time steps. The problem is less severe at the earlier time steps, where the bubbles are well defined and we need to only focus on robust algorithms for extracting them. As the data is stored on tape, it can be time consuming to access the data. This precludes multiple passes through the data set and alternative approaches have to be considered, first to define the bubbles appropriately, and then to extract the necessary information from these multi-terabyte data sets.

3 Related Work

The analysis of bubble and spike structures in Rayleigh-Taylor and Richtmyer-Meshkov instabilities has been the subject of considerable prior work in the fluid mechanics community. We next review this work briefly, as well as related work in the analysis of coherent structures.

The early work in the evolution and structure of the bubbles and spikes in the Rayleigh-Taylor mixing front was pioneered by Sharp and Wheeler [14]. They proposed a model of bubble amalgamation, where an initial period of linear growth was followed by a nonlinear stage, where the bubbles of different sizes rose through the denser fluid at different velocities. Based on experimental evidence which suggested that a large bubble expanded and moved ahead of its smaller neighbors, which shrank and were washed downstream (a process referred to as “bubble merger”), they proposed a set of rules governing the bubble merger.

This early model has since been refined to minimize discrepancies with experiments as well as 2-D and 3-D, high-resolution, multi-mode simulations. Several authors have contributed to this work. For example, Shvarts et al. [15] observed that in 3-D, as the bubbles grew, they became rounder, losing all memory of the anisotropy of the initial perturbation, where the bubbles had contorted worm-like shapes with widely varying orientations and aspect ratios. Their simulations were on relatively small 3-D grids of size $48 \times 48 \times 96$. The different phases of the evolution of the bubble front were described by Oron, Alon, and Shvarts [13] who tested their model using a full, multi-mode, numerical simulation, starting with the initial condition of 50 bubbles. Their visual illustration of the evolution of the bubble front over time is a simple way to identify the bubbles which have merged, at least in two dimensions. They also compare the number of rising bubbles obtained from the simulations with those obtained from the models. A different view of the bubble merger process based on renormalization group dynamics and an approximation of the bubbles as elliptically-tipped cylinders is discussed in Cheng, Glimm, and Sharp [4]. Their results show that a non-uniformly distributed bubble radius enhances the bubble merger rate and increases the interpenetration between the fluids.

A comparative study of the different high-resolution simulations is described in Dimonte et al. [6]. The authors compare the results from various simulations, some on grids as large as $256 \times 256 \times 512$. The results from these simulations are also compared with experiments, such as the ones done using the Linear Electric Motor by Dimonte and Schneider [7].

The studies mentioned above have focused on the growth of the mixing layer starting with different initial perturbations and with different ratios of fluid densities. While several of them do discuss the extraction of quantities such as the bubble diameters, they unfortunately do not provide an algorithm which defines the extent of a bubble, especially once the bubbles have started to merge. This made it difficult to use a physics-based definition of the bubbles in our work. In addition, few papers discuss the issues related to the processing of large amounts of data from high-resolution, 3-D simulations.

Coherent structures in neutral fluids have also been analyzed using techniques from image processing and data mining. One technique that has been extensively used is wavelets [8, 9]. As wavelets operate on different scales, the information they extract from the data is ideal for identifying structures at different scales in problems such as turbulent flow. For example, Farge and Schneider [10] describe a nonlinear procedure to filter wavelet coefficients to separate the coherent vortices from the background flow. In a similar manner, Siegel and Weiss [16] use wavelet packet technology, again in the context of fluid flow, to separate signals into coherent and non-coherent parts. A slightly different approach is taken by Hangan et al. [12] who combine wavelets with the more traditional template matching approach from pattern recognition to match patterns at different scales.

Data mining techniques have also been used to identify and track coherent structures. For instance, Ferre-Gine et al. [11] use a fuzzy ARTMAP neural network to identify eddy motions in a turbulent wake flow, using data from a wind tunnel. Banerjee, Hirsch, and Ellman [1] use a tracking method based on decision trees to identify related vortices in different time steps, instead of using a traditional tracking method with computationally expensive heuristics to track the objects through all intermediate time steps.

Other work for the analysis of coherent structures includes those motivated through visualization, such as the visiometrics approach of Zabusky [18] and the references therein.

4 Description of the Data

We next describe how the LES and DNS data sets are stored in multiple files at each time step. This, along with the size of the data and the lack of precise definitions for bubbles and spikes, drives the algorithms we can use in our analyses.

4.1 Data from the large-eddy simulation

As mentioned earlier, the LES data is an 1152^3 data set, containing over 1.5 billion grid points. The initial interface between the two fluids is at $z = 612$, assuming $z = 0$ as the bottom boundary of the domain. This simulation was run on the LLNL Linux clusters, ALC and MCR, using 1728 Pentium 4-processors, each running at 2.4GHz. The entire simulation took 30 CPU days to complete. The 1152^3 data is partitioned among the 1728 processors in vertical columns of size $24 \times 32 \times 1152$ in the x, y, and z directions, respectively. This partitioning results in a 48×36 grid of processors, which is correspondingly the grid of columns that would reconstruct the full data. Each processor in the 48 × 36 grid is given a processor id in a row major order starting from the bottom left corner of the grid.

In this simulation, there are 759 time steps including the initial starting state. For each time step, results from the simulation are stored in directories with the prefix "dump" followed by a four digit number of the time step with leading zeros. Within each dump directory, there are 1728 files representing the column output from each processor. These vertical column results are stored with the prefix 'p' followed by a four digit number for the processor id with leading zeros.

Each column stores the data in an identical form. Recall that for the LES simulation there are seven variables output at each grid location. All data files begin with an integer record size at the beginning of the file. This is the size of the data in bytes, which is exactly 24,772,608 bytes, equal to seven, single-precision, floating-point values for the 24 × 32 × 1152 grid points in a column. This record size does not include the integer value itself. After the record size, the x velocity values for all grid points are stored, followed by the y velocity, the z velocity, the density, the pressure, the viscosity, and the diffusivity. The file ends with the integer record size. All data values are stored in the little-endian binary format.

4.2 Data from the direct numerical simulation

The DNS data is a 3072^3 data set, containing over 28.9 billion grid points. This simulation was run on the BG/L system at LLNL, which is based on the PowerPC 440 700MHz chips, rated at 2.8 GFlop/s. The number of processors used varied from 16K at the early time steps to 64K at later time. The simulation took 17 days to complete, accounting for a total of 2303 single-CPU years. Unlike the LES data, the DNS data set does not have the same size for all time steps. The z-dimension is varied over time to accommodate the region around the interface which grows as the two fluids mix. At the early time steps, the height of the mixing layer is small, and only a small portion of the entire z-dimension is necessary. At later time, when the mixing layer is a large fraction of the z-dimension, all 3072 grid points along z are used. The values of the z-dimension used for all time steps are given in Table 1. As a result of this variation in the z-dimension, the initial interface is computed as a function of the number of grid points in the z-direction. Assuming $z = 0$ as the bottom boundary of the domain, the initial interface is at $z = 17 \times nz/32$ where nz is the number of grid points in the z-direction at any time step.

As in the LES data, the DNS data is partitioned in vertical columns equal in number to the number of processors used. For the DNS data set however, the time steps have been computed using either 16384, 32768, or 65536 processors. As a result, the x- and y-dimensions of the columns also vary with the number of processors used. The details of the processor partitioning are summarized in Table 1.

The simulation contains 248 time steps including the initial starting state. The results for each time step are stored in a directory with the prefix "viz" followed by a four digit number of the time step with leading zeros. Within each directory, there are either 16384, 32768, or 65536 files corresponding to the column output from each processor. These columns are stored with the prefix 'p' followed by a six digit number of the processor id with leading zeros. The processor-ids start at zero and are numbered consecutively. Unlike the the LES data set, the processors for the DNS data are ordered in column-major order starting from the bottom left

Range of time steps	Column sizes $x \times y \times z$	Processor distribution $px \times py \times pz$
0 - 26	$24 \times 24 \times 256$	$128 \times 128 \times 1$
27 - 53	$12 \times 24 \times 512$	$256 \times 128 \times 1$
54 - 89	$12 \times 24 \times 1024$	$256 \times 128 \times 1$
90 - 122	$12 \times 24 \times 1536$	$256 \times 128 \times 1$
123 - 127	$12 \times 24 \times 2048$	$256 \times 128 \times 1$
128 - 145	$12 \times 12 \times 2048$	$256 \times 256 \times 1$
146 - 162	$12 \times 12 \times 2560$	$256 \times 256 \times 1$
163 - 247	$12 \times 12 \times 3072$	$256 \times 256 \times 1$

Table 1: DNS data set: processor partitioning and column sizes.

corner of the grid.

As mentioned earlier, the DNS data set contains the five variables - the x, y, and z velocities, the density, and the pressure. Within each vertical column, there are five records, each containing one of the variables in the above order. For each record, there is a preceding and trailing integer value indicating the record size. This record size will vary depending on the number of grid points within the column, as listed in Table 1. Although the columns are ordered in a column-major order, the data within the columns are ordered in row-major order. The actual numerical values for all grid points are written in single-precision floating point values in big-endian binary format.

5 Exploratory Analysis of the Data

The first step in our analysis was to explore the data at select time steps so we could propose a definition of a bubble and determine possible ways of identifying the bubbles and extracting the necessary statistics for them. The simulation data is on a regular Cartesian grid in three dimensions, with uniform spacing in the x, y, and z directions, and $\Delta x = \Delta y = \Delta z$. This allows us to consider the data as an "image" in three dimensions and apply traditional image processing techniques after suitably modifying them to handle the third dimension.

Our exploratory analysis of the 3-D data focused on a subset of the data from the LES simulation. We considered the grid points in the central 6×6 columns at every 50-th time step, starting with the initial time. There were 16 such time steps, with the seven variables at each time step available at $144 \times 192 \times 1152$ grid points. The choice of the 6×6 columns was large enough to enable us to follow the bubbles as they evolved over the 759 time steps, yet small enough to allow easy experimentation.

Figures 2 and 3 show a slice of the density variable taken at $x = 0$ in the 6×6 columns. Figure 2 is for time steps 0 through 450, in steps of 50. The images are 192×400 and have been cropped in the z direction to show the detail around the interface. Figure 3 is for time steps 500 through 750, in steps of 50. These images are 192×700 and have been cropped in the z direction.

The convention used in these and other images in this report is from the image-processing community, where white indicates the higher density fluid and black is the lighter density fluid, that is, lower valued pixels are darker. This is the opposite of the convention often used in the fluid mix community, where the denser fluid is indicated in black. We also apply all image processing algorithms to floating point data in contrast with traditional image processing techniques which operate on 8- or 16-bit integers.

We observe the following behavior of the bubbles. At the very early time steps (until time step 100 or so), the bubbles are small and relatively well defined. As they grow, they appear to merge, at least in these example slices through the data. Their structure also becomes less well defined, that is, it is no longer possible to look at a slice and clearly identify the extent of a bubble in that slice. At later times, around time step 400, there are two bubbles which stand out, though their boundaries are not as well structured as at the early time steps. At time step around 600, the two bubbles appear to merge, starting near the top. The width of the slice in this example is insufficient to capture the full bubble at later time.

It is important to note that the images in Figures 2 and 3 show a 2-D slice through the bubble structure at $x = 0$ of the 6×6 columns at the center of the domain. The real structure of the bubble can be understood only by considering several consecutive slices through the columns, as shown later in Figure 5 in Section 6.

These images illustrate several of the challenges in the analysis. The initial scale of the structures of interest is quite small. The bubbles at early time are just a few grid points (or pixels) above the interface between the dark (lighter fluid) and light regions (heavier fluid). At later time, say around time step 300, structures of this size are mainly noise as the structures of interest are much larger in size. We also observe that at later time, as the bubbles merge, it is difficult to identify the boundary of each bubble, giving rise to questions on what exactly is a bubble. For example, if two bubbles meet at a grid point (or a few grid points), do we count them as one? Or, should they be counted as one bubble only when they have merged "substantially", however we choose to define this? Further, if a bubble ceases to grow, and slowly "disappears" as the bubbles around it grow, at what point do we stop counting the bubble which disappears? We will revisit these and other challenges as we discuss our approach to the analysis.

6 Analysis Part I: Identifying the Bubbles/Spikes in 3-D Data

Our first task in the analysis was to identify the bubbles in the 3-D data. As mentioned earlier, we can consider the data as a 3-D "image" and apply traditional image processing techniques, such as segmentation, which have been extended to three dimensions. We considered two approaches to identify the 3-D boundary of the bubble - a 3-D version of the Canny edge detection technique [3] and a 3-D version of a traditional region growing method [17], which was modified to incorporate the constraints we thought were suitable for defining a bubble.

Our choice of simple segmentation techniques for the identification of the bubbles in the 3-D data was driven by the size of the data and its complexity. More complex techniques such as active contours (either snakes or level sets) are also a possibility. However, they are iterative

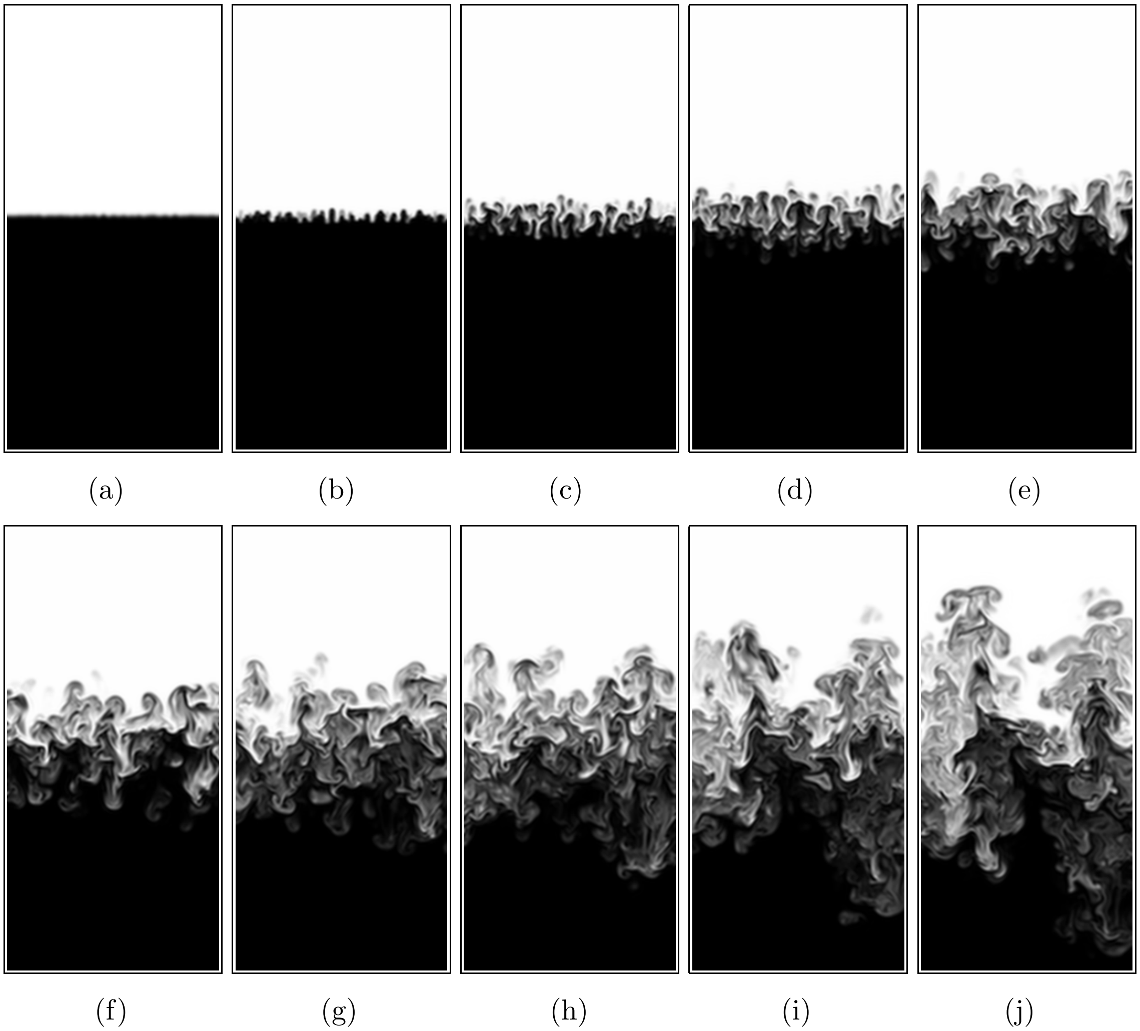


Figure 2: Two-dimensional slices through the density variable at $x = 0$ for the central 6×6 columns of the LES data at time steps 0 through 450 in steps of 50. The images are 192×400 and have been cropped in the z direction to show the detail around the interface. The heavier fluid on top appears in white, while the lighter fluid at the bottom is in black. The image widths are one-sixth of the full simulation domain.

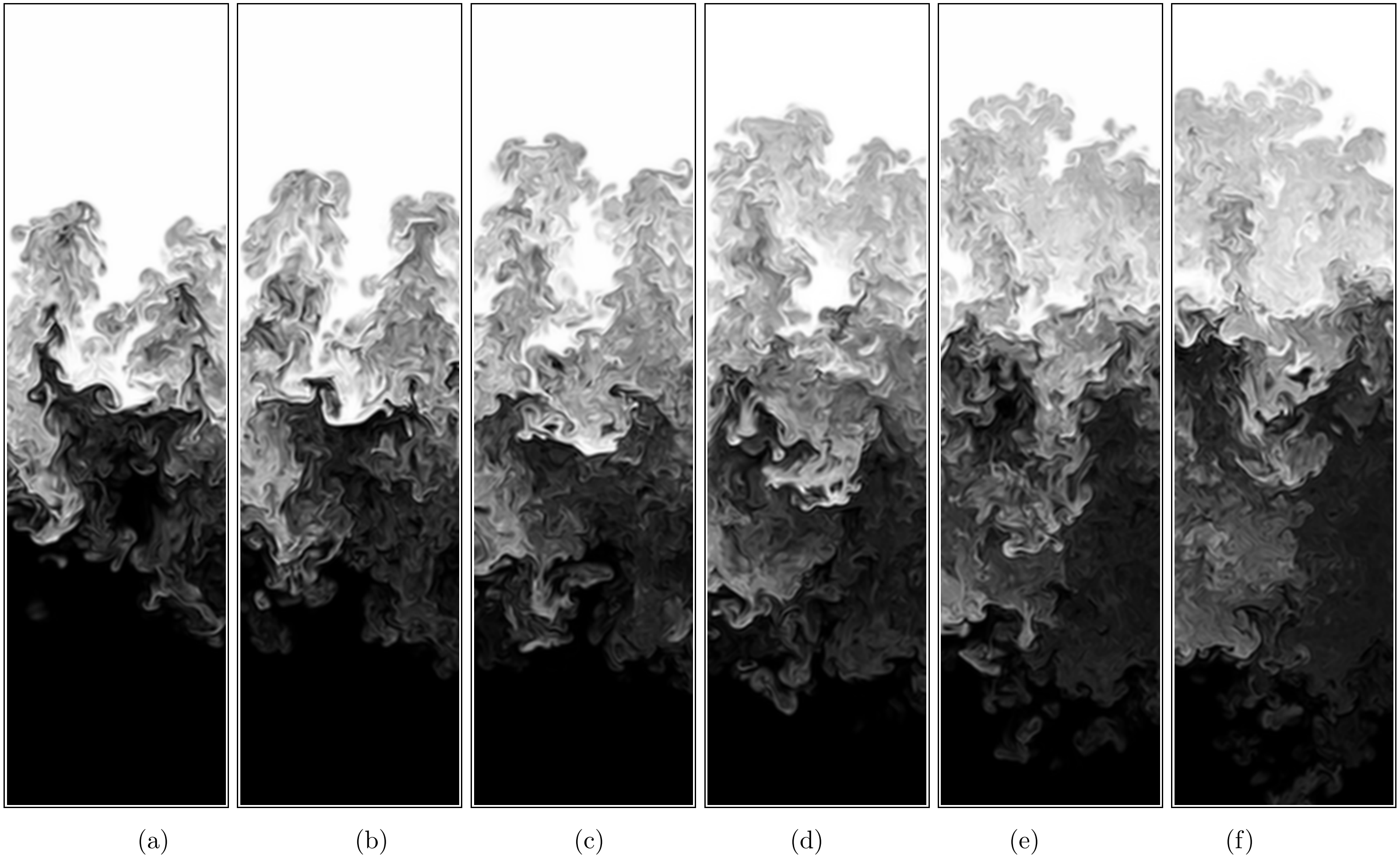


Figure 3: Two-dimensional slices through the density variable at $x = 0$ for the central 6×6 columns of the LES data at time steps 500 through 750 in steps of 50. The images are 192×700 and have been cropped in the z direction to show the detail around the interface. The heavier fluid on top appears in white, while the lighter fluid at the bottom is in black. The image widths are one-sixth of the full simulation domain.

in nature, require the setting of several parameters, and are complex to program, making them a poor choice for this problem.

Another issue which influenced our choice of segmentation algorithms was the distribution of the data. As described earlier, the data is stored as several columns for each time step. Ideally, we were interested in segmentation techniques which could be applied to a column of the data at a time, and the results patched together for the segmentation of the entire cube. Iterative techniques such as active contours, would have required a parallel implementation to operate on the entire cube, which was another reason why they were not considered. We discuss the issues with the parallel application of the Canny edge detector and the region growing technique further in Section 6.4.

Our work on the 3-D segmentation focused mainly on the density variable. Though other variables are also available for the LES and DNS datasets, a visual inspection of 2-D slices of the data indicated that it was easiest to identify the boundary of the bubbles in the density variable. The only other variable which was equally effective was the magnitude of the vorticity; however, this option was not considered as it required additional computation to first calculate the vorticity.

6.1 3-D Canny edge detector

The Canny edge detector [3] is an optimal edge detector and was designed to have a low error rate. It misses few edge pixels, detects few false-positives, achieves good localization of edges, and minimizes the number of responses to a single edge. An implementation of the Canny edge detector consists of the following steps:

1. Step 1:

   Smooth the data with a Gaussian filter. The σ of the Gaussian essentially determines the scale of the structures in the data which will be smoothed away and whose edges will therefore not be detected. Unlike the original Canny detector which suggested using a range of scales, we use just a single scale in our analysis.

2. Step 2:

   Compute the magnitude and direction of the gradient at each grid-point (i.e. pixel) using finite difference approximations to the partial derivatives, for example, through the Sobel edge detector [17].

3. Step 3:

   Apply non-maximal suppression to the gradient magnitude. A pixel is retained if the gradient magnitude of the two neighbors on either side of the pixel in the direction of the gradient is smaller than the gradient magnitude at the pixel. This effectively thins the edge image.

4. Step 4:

   Perform hysteresis thresholding on the edge image to detect and link the edges. This thresholding uses two thresholds. All pixels with gradient magnitude larger than the high threshold are kept, while those smaller than the low threshold are removed from

> the final image. Pixels with gradient magnitude between the low and high thresholds are kept only if they are connected (recursively) to a pixel with magnitude greater than the high threshold. Relative to the use of a single threshold, the hysteresis thresholding fills in some of the gaps in the edge image without introducing false-positive edges.

We extended the Canny edge detector, which was originally proposed for 2-D images, to 3-D data. In both cases, it has four parameters - the σ for the Gaussian, the size in pixels for the Gaussian filter, and the two thresholds for the hysteresis thresholding. The size of the filter can be set automatically given a value of σ.

We have found the Canny edge detector to be very effective over a range of problems due to its optimal edge-detection properties, the small number of parameters, and the robustness of the results to the choice of parameters. However, like other edge detectors, it can result in incomplete edges, especially in regions of the image where the gradient is not high. These gaps in the edges are often large enough that they cannot effectively be filled in by an edge-filling technique [17].

Figure 4, panel (a), shows the results of the Canny edge detector using the density variable at time step 350. The image is of the $x = 1$ slice from the central 6×6 columns. Only the region around the fluid interface has been shown for clarity instead of the entire column. The left panel (a) indicates the edges obtained after applying a 3-D version of the Canny edge detector to the $144 \times 192 \times 1152$ pixels of the 6×6 columns using no Gaussian smoothing, and thresholds $t_{lo} = 0.1$ and $t_{hi} = 0.2$. The edge pixels are indicated in red and superposed on the original data.

6.2 3-D region-growing segmentation

To address the problem of gaps in the edges found by the Canny edge detector, we also considered a region-growing approach to the identification of bubbles in 3-D. Starting with an initial pixel, these approaches grow a region by adding neighboring pixels which satisfy certain constraints. As only neighboring pixels are connected, we obtain a closed contour around the objects of interest. This approach is dependent on the choice of constraints, the starting pixels, and the order in which pixels are selected for consideration for merging with an appropriate region.

Figure 4, panel (b) shows the results of the 3-D region growing technique for the same slice as in panel (a), which was described earlier. The pixels in pink highlight the bubble/spike pixels. The idea behind our implementation of the region growing algorithm is simple - essentially, we start growing a region each from the top and the bottom of a column. The top region grows downwards. A new pixel is added to the region provided it is 6-connected to the region and its intensity, along with the intensities of all of its 6 neighbors, is greater than a high threshold. Likewise, the bottom region grows upwards. A new pixel is added to the region provided it is 6-connected to the region and its intensity, along with the intensities of all of its 6 neighbors, is lower than a low threshold. For this example, the high and low thresholds were set to be 2.8 and 1.2, respectively. The 6 neighbors considered are 1 pixel away in the x, y, and z direction from the pixel under consideration. More stringent constraints using 18 neighbors, as well as all 26 neighbors, were also tried without a substantial difference in the results. The only noticeable difference was “blockier” boundaries of the bubbles and spikes due to the “box-like” mask being used.

Figure 5 shows the results of the region-growing technique on every alternate slice for the central 6×6 columns, starting from the slice at $x = 1$ and ending at the slice at $x = 19$. It more clearly shows the complex 3-D structure of the bubbles and spikes.

6.3 Converting the 3-D data to 2-D

The two images in Figure 4, as well as the sequence of images in Figure 5 indicate that, even though the region growing and edge detection techniques help to identify the boundary of the bubbles/spikes in 3-D, it is non-trivial to extract the required statistics from this information. To reduce the data further, we considered the top and bottoms views of the 3-D data. The images formed by taking the height/depth (from the initial interface between the two fluids) of each pixel at the bubble/spike boundary are shown in Figure 6. For both 3-D segmentation algorithms, we consider the bubble height (spike depth) to be the height (depth) of the first edge pixel encountered along the positive (negative) z axis as we move towards the fluid interface starting at the top (bottom) of the 6×6 columns. Note that there may be more than one edge pixel along z for each (x, y) location. For the bubbles, we select the one which is closest to the top of the cube and similarly, for the spikes, we select the one which is closest to the bottom of the cube.

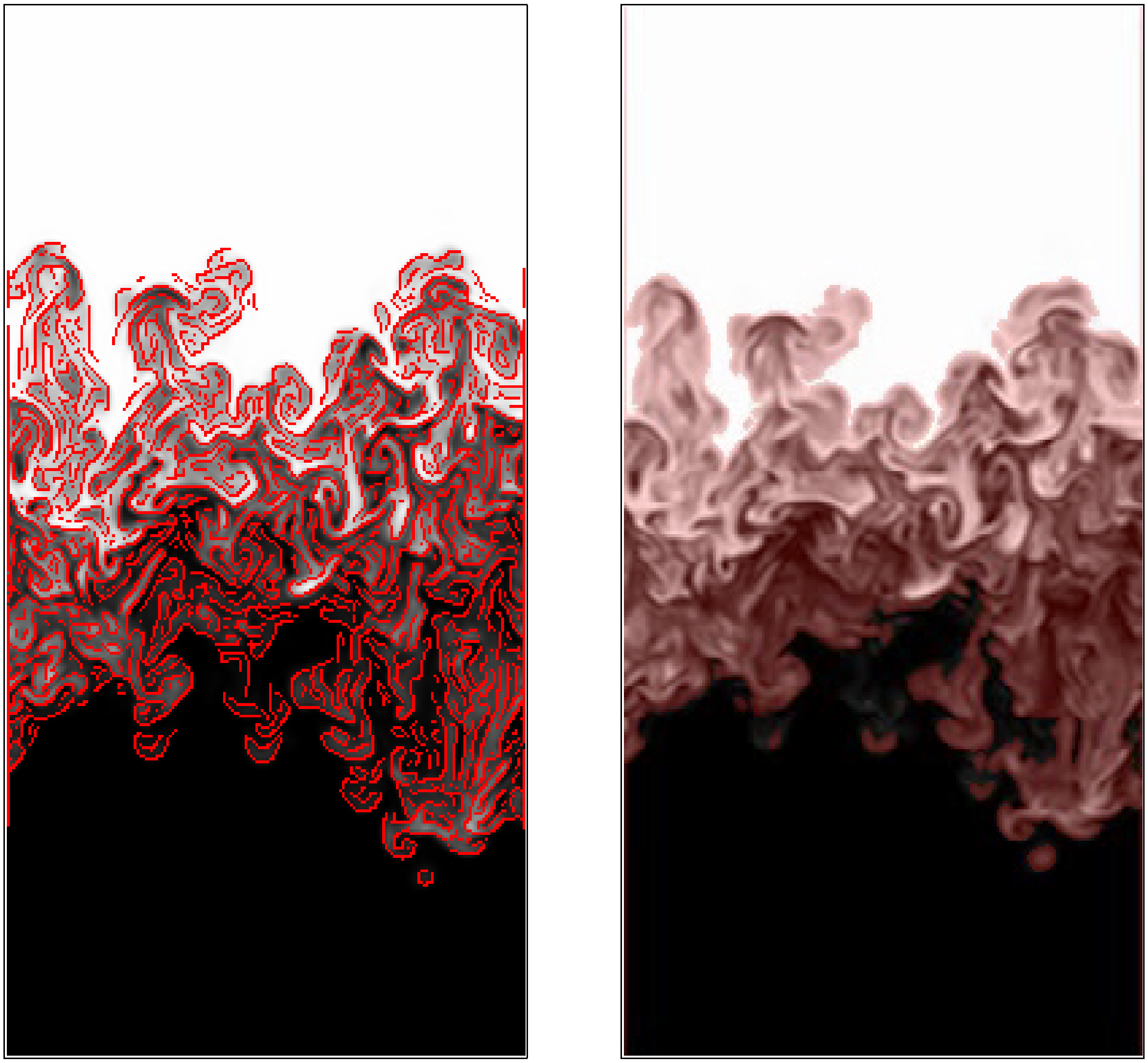
The images in Figure 6 have been normalized so that the highest pixel in the bubble images and the lowest pixel in the spike images, both measured from the fluid-mix interface, correspond to a pixel intensity of 255. This top view of the bubbles implies that brighter pixels are closer to the top, or further from the fluid interface. Similarly, the bottom view of the spikes implies that the brighter pixels indicate spikes which go deeper into the lighter fluid.

As can be expected from the images in Figure 4, the Canny edge detector, due to the gaps in the edges, could give rise to bubble/spike boundaries which are not nearly as continuous in 3-D as the ones obtained using the region-growing approach. This can be clearly seen in the two bright bubbles near the bottom of Figure 6 (a). The surfaces of these bubbles have darker spots, indicating that some locations have a lower height than the immediate surrounding regions. Further, a closer look at the images from the 3-D Canny edge detector and the 3-D region growing approach indicates that the boundary of the bubbles and spikes in 2-D is less smooth when the edge detector is used. Aside from these differences, we observe that the top (bottom) view of the bubbles (spikes) obtained from the two algorithms are very similar, indicating that our approach is relatively independent of the choice of algorithms used in the 3-D segmentation.

In our experiments with the central 6×6 columns from every 50-th time step of the LES simulation, we found that we could use a fixed set of parameters for the Canny edge detector (no Gaussian smoothing, $t_{lo} = 1.0$ and $t_{hi} = 2.0$) for all the time steps. In contrast, the two parameters for region growing (the thresholds for the bubbles and spikes) had to be modified suitably to identify the bubbles and spikes correctly at later time steps. We shall address this issue of the choice of parameters later in Section 6.6.

6.4 Parallel application of segmentation algorithms

The size of the 3-D data for both the LES and the DNS simulations, and its distribution among many files, require that we choose algorithms which have low computational complexity and are amenable to parallelization. The former constraint certainly holds for both the Canny



(a)

(b)

Figure 4: Defining a bubble in 3-D. The image is the $x = 1$ slice from the central 6×6 columns of time step 350 of the LES data. The images are cropped in the z direction to show only the region around the interface. (a) The results of the 3-D Canny edge detector (in red) superposed on the original 3-D image. (b) The results of the 3-D region growing segmentation (in pink) superposed on the original image.

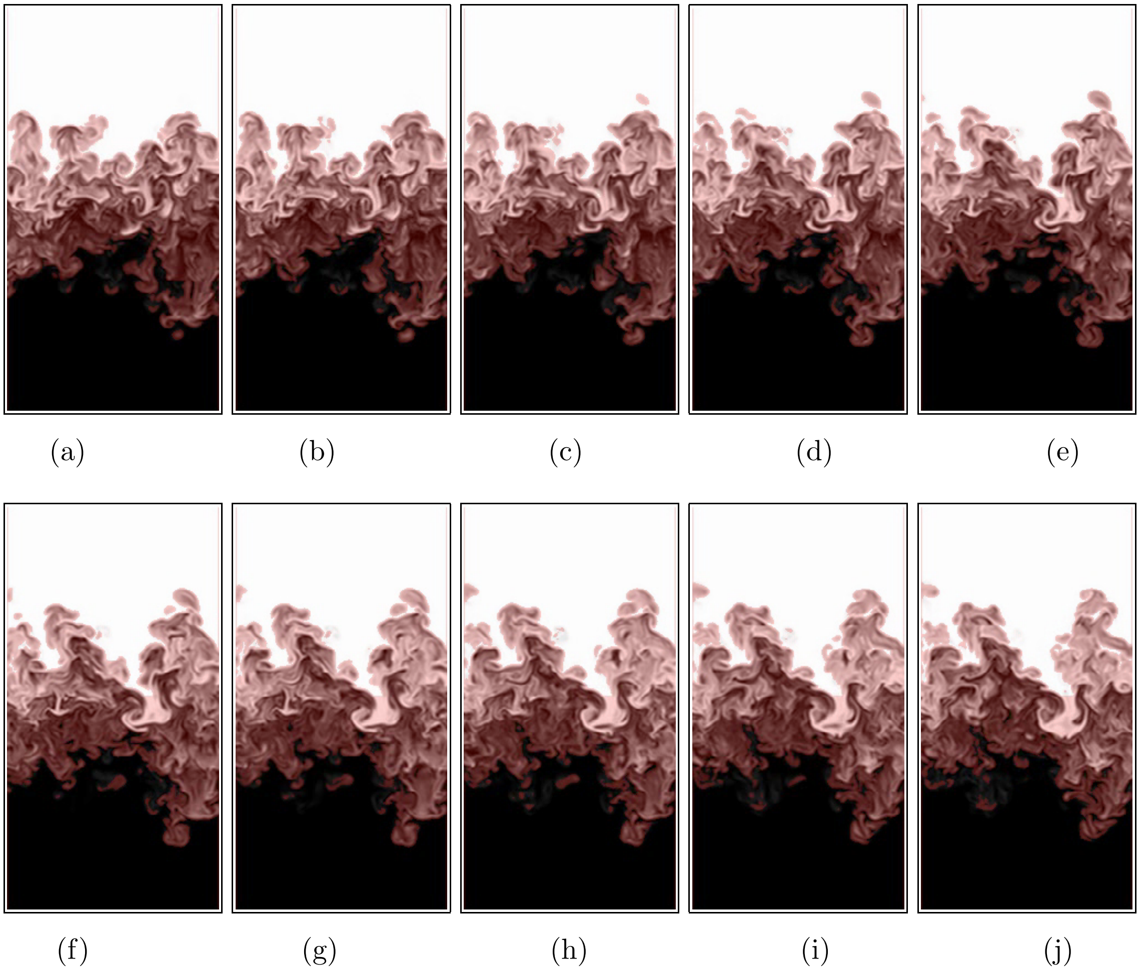
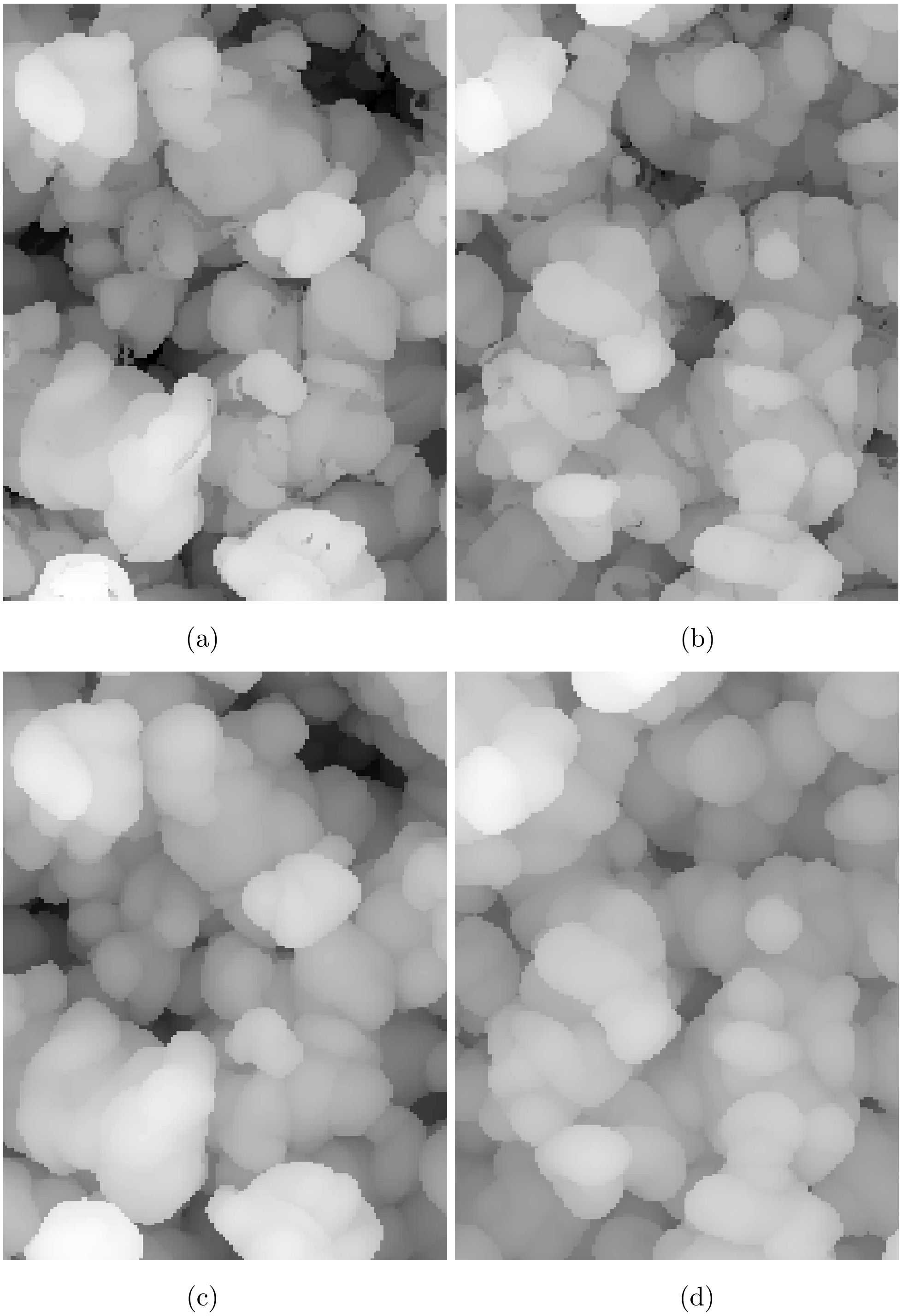


Figure 5: The complex 3-D structure of bubbles and spikes illustrated using the 3-D region-growing technique. The figure shows every alternate slice, starting from the slice at $x = 1$ (panel (a)) and ending at the slice at $x = 19$ (panel (j)) of the central 6×6 columns of the LES data at time step 350.



(a) (b)

(c) (d)

Figure 6: The top (bottom) view of the bubbles (spikes) for the central 6 × 6 columns of the data at time step 350. (a) and (b): The bubbles and spikes obtained using the 3-D Canny edge detector. (c) and (d): The bubbles and spikes obtained using 3-D region-growing segmentation.

and the region growing techniques. We next explore the issue of parallel implementation and application of these techniques.

To process the data in each column, both the 3-D Canny edge detector and the 3-D region-growing approach require the pixels in the neighboring columns. The number of neighboring columns depends on how the structures are distributed among the processors. Our application of the Canny edge detector does not use any Gaussian smoothing, else it would have required each column to be expanded by a suitable number of pixels from the neighboring columns so that the Gaussian could be applied to all pixels in the column. The Sobel edge detector requires the column to be expanded by 1 along the four sides in the x and y direction. However, the hysteresis thresholding could require information from several adjacent columns to handle the edge pixels with intensities between t_{lo} and t_{hi}. Consider a very simple case of a string of pixels in a straight line, whose intensities fall in this category. Let the two end pixels of this line have intensity greater than t_{hi}. If this line is distributed across two processors, each of which has one end-point, then hysteresis thresholding done separately on each processor will give results identical to the results on one processor. However, if the line is distributed among three processors, only two of which have the end points, then the line pixels in the processor with no end point will not be detected as an edge. This is because none of these pixels exceeds the t_{hi} threshold. Addressing this problem would require several passes through the data during hysteresis thresholding, with communication of the boundary values after each pass, until all edge pixels have been appropriately identified. This can be quite cumbersome if, instead of reading all the files at a time step back onto the parallel system, we do the edge detection by processing a single column (or a bunch of columns) at a time, suitably padded with pixels from neighboring columns. An alternative solution would be to use a single threshold instead of hysteresis thresholding. However, this would lead to additional gaps in the edge image, aggravating the problem of gaps in the bubble surface seen in Figure 6.

The parallelization of the 3-D region growing poses problems of its own. We illustrate this using a 2-D slice through the data in Figure 7, where the left half of the image is assigned to one processor and the right half to another processor. Consider the region growing process for the bubbles on the left half which starts from the top of the image. There is a part of the heavier fluid (in white, towards the right edge of the left half of the image) which cannot be reached by a region growing from the top. The only way this can be reached is through the processor on the right after it has completed its region growing and communicated the results to the processor on the left. This technique therefore has the same drawback as the hysteresis thresholding in the Canny edge detector, requiring multiple passes to obtain results identical to a single processor.

However, when we consider the follow-on step to the 3-D segmentation, namely, looking at the bubbles (spikes) from the top (bottom) of the cube, we realize that we do not require the 3-D segmentation to provide identical results for the single and multiple processor cases. We only require that the results after the follow-on step be independent of the number of processors used. For the region growing approach, this means that we do not need to identify the small region of the heavier fluid in the left processor which is reachable only from the right processor, as this region does not appear in the top down image. Therefore, the region-growing approach can be implemented with minimal communication, requiring padding only a pixel wide from neighboring processors. The top-down (bottom-up) image for each processor (after removal of the padding), when patched together across all processors, will give the resulting

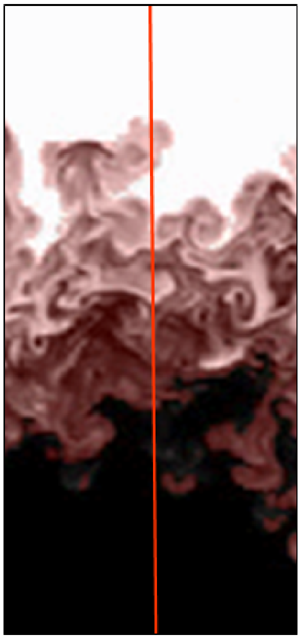


Figure 7: Problems with parallelization of the 3-D region growing method illustrated using a sample slice. The line in red is the processor boundary, with the left half of the image assigned to one processor and the right half to another.

image for the bubbles (spikes) for the entire cube.

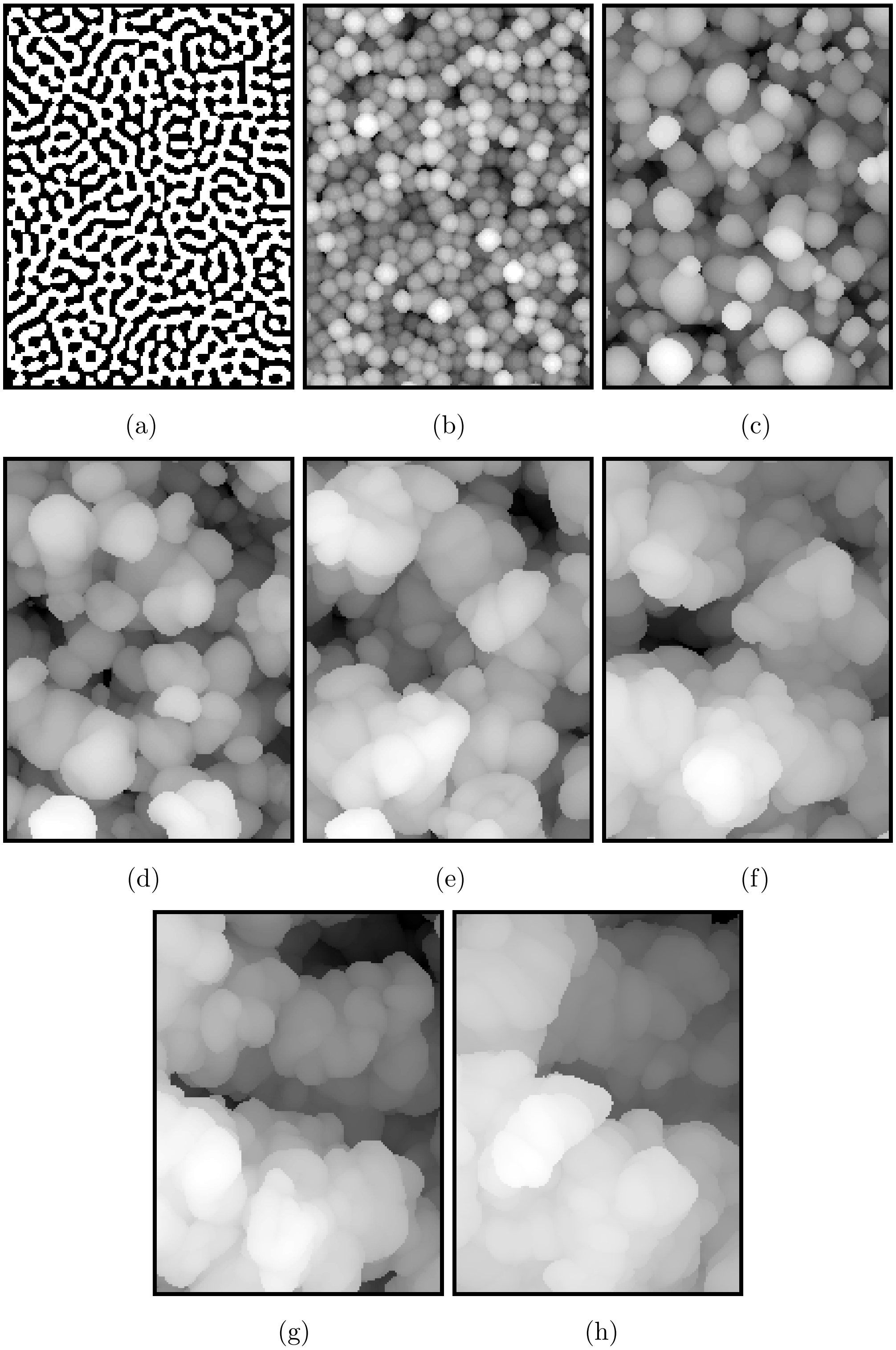
6.5 2-D images of bubbles and spikes

In the rest of the paper, we shall focus on the 3-D region-growing approach and the 2-D images obtained using the top (bottom) view of the bubble (spike) surface. We shall not consider the 3-D edge detection approach using the Canny technique in light of the difficulties in parallelization as well as the lower quality of the resulting 2-D images.

The 2-D images of bubbles and spikes for the central 6×6 columns, for every 100-th time step for the LES simulation are shown in Figures 8 and 9. We refer to these images as the height-depth maps (HDM) as they reflect the height (depth) of bubbles (spikes) from the fluid interface. Recall that these were obtained through region growing using the density variable.

Figures 10 and 11 show the values of the other variables at the bubble height and spike depth, respectively, for the LES data. These variables include the pressure, the velocities in the x, y, and z directions, and the magnitude of the $x - y$ velocity. We included the HDM to complete the set and the magnitude of the $x - y$ velocity because a visual analysis of the x and y velocities indicated that combining the two could potentially help in the identification of the bubbles. We explore this idea further in Section 7.2. Figures 12 and 13 show similar images for the central 300×300 pixel column from the DNS data. The thresholds used for generating these images for both the LES and the DNS data are described in Section 6.6.

The 2-D images of the bubble height-depth maps and the different variables at the bubble

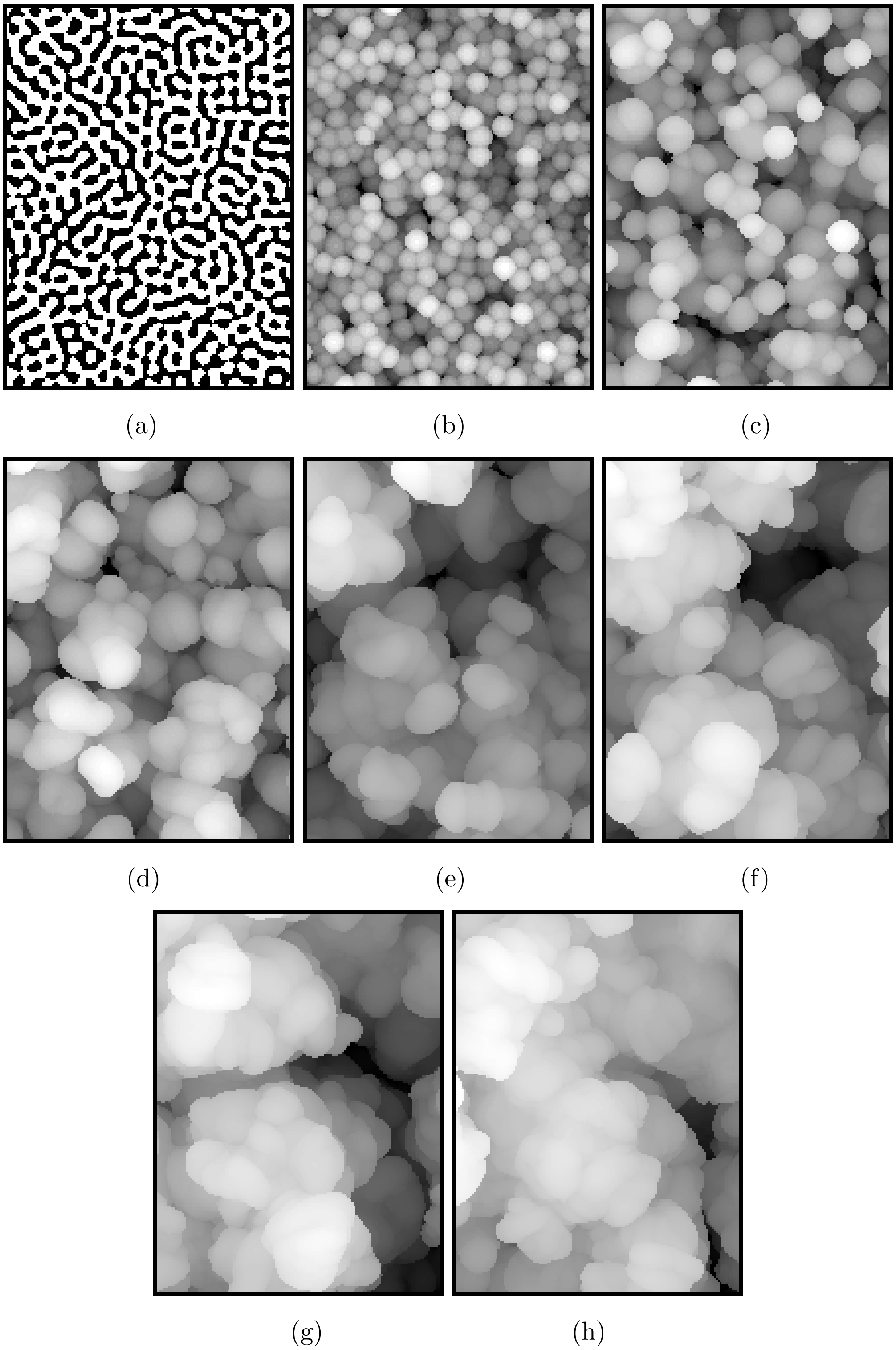


(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 8: The top view of the bubbles obtained using 3-D region-growing for the central 6×6 columns of the LES data at every 100-th time step, starting at time step 0. Each image is normalized so the intensity values lie between 0 and 255. The brighter pixels are closer to the top of the cube.



(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 9: The bottom view of the spikes obtained using 3-D region-growing for the central 6×6 columns of the LES data at every 100-th time step, starting at time step 0. Each image is normalized so the intensity values lie between 0 and 255. The brighter pixels are closer to the bottom of the cube.

boundaries, Figures 10 through 13, further illustrate the growth of the bubbles and spikes as well as some of the challenges. For the LES data, these images clearly show how the bubbles start out small and unstructured at time step 0 (the initial perturbation) and become rounder (in the top view) at time step 100. During this process, the bubbles, which are initially isolated, grow to fill the entire 2-D image, until around time step 100, we see that some bubbles have grown faster than the others (indicated by a brighter intensity) and cover parts of the top view of their neighbors. We also observe that until time step 100, the bubbles are very similar in size to each other. However, at time step 200, there is a large variation in the bubble sizes, with some bubbles staying the same size as at time step 100, while others have grown to more than twice their size. The bubbles are still distinctly circular, though some cover others substantially. At time step 400 onward, as the bubbles merge, they lose their circular shape.

Additional details are visible when we create a movie made of these images using all the time steps. We observe that the bubbles which grow, start out small, relative to their neighbors. As they grow, they barrel through the neighboring bubbles, gaining in height. After a while, the drag which is proportional to the size of the bubble, slows down its growth. At this point, a different bubble starts to grow, and after a while slows down as well. During this growth, it may also merge with neighboring bubbles and sometimes even split. When it merges, the two or more individual bubbles do not grow as one, and appear to maintain their individual growth rates, and may even grow or shrink independent of the other. Though they appear as a single bubble, their identities remain distinct for a while.

These images also illustrate the challenges in identifying the bubbles. At early time, we can clearly identify the extent of each bubble. As the bubbles merge, the surface of the merged bubble becomes more “bumpy”. These bumps eventually merge into the rest of the bubble in that they are no longer discernible as separate entities. The “bumpiness” of the surface makes it difficult to identify the extent of each bubble as one part of the bubble may be clearly distinguishable from the background, but the other part may not. The merging and splitting also complicate statistics such as the count and the size of the bubbles as it is not clear when in the process of merging or splitting, we start to, or cease to, count more than one bubble as one.

6.6 Choice of thresholds in 3-D region-growing segmentation

As mentioned previously, the 3-D region growing technique for obtaining the height-depth map of the bubbles and spikes uses thresholding of the density variable to ascertain the location of the bubble and spike boundaries. Selecting this threshold is not a simple task, especially when we consider the variation in the data values over time, as well as the changes in the scale of the structures of interest. In this problem, the composition of the fluids at the boundary of the bubbles (spikes) changes as a function of time as the two fluids of differing densities mix under the influence of gravity. Initially, we can use a large range of threshold values to correctly identify a bubble or a spike. This is because there is a large gradient at the bubble boundary, and many threshold values greater than 1.0, but less than 3.0 will identify the boundary correctly. As the two fluids mix further, their density is diffused and the range of density values that can accurately describe the surface of the bubbles and spikes decreases. We illustrate this later on in Section 9.

Our initial experiences with identifying the bubbles in 3-D using the region growing tech-

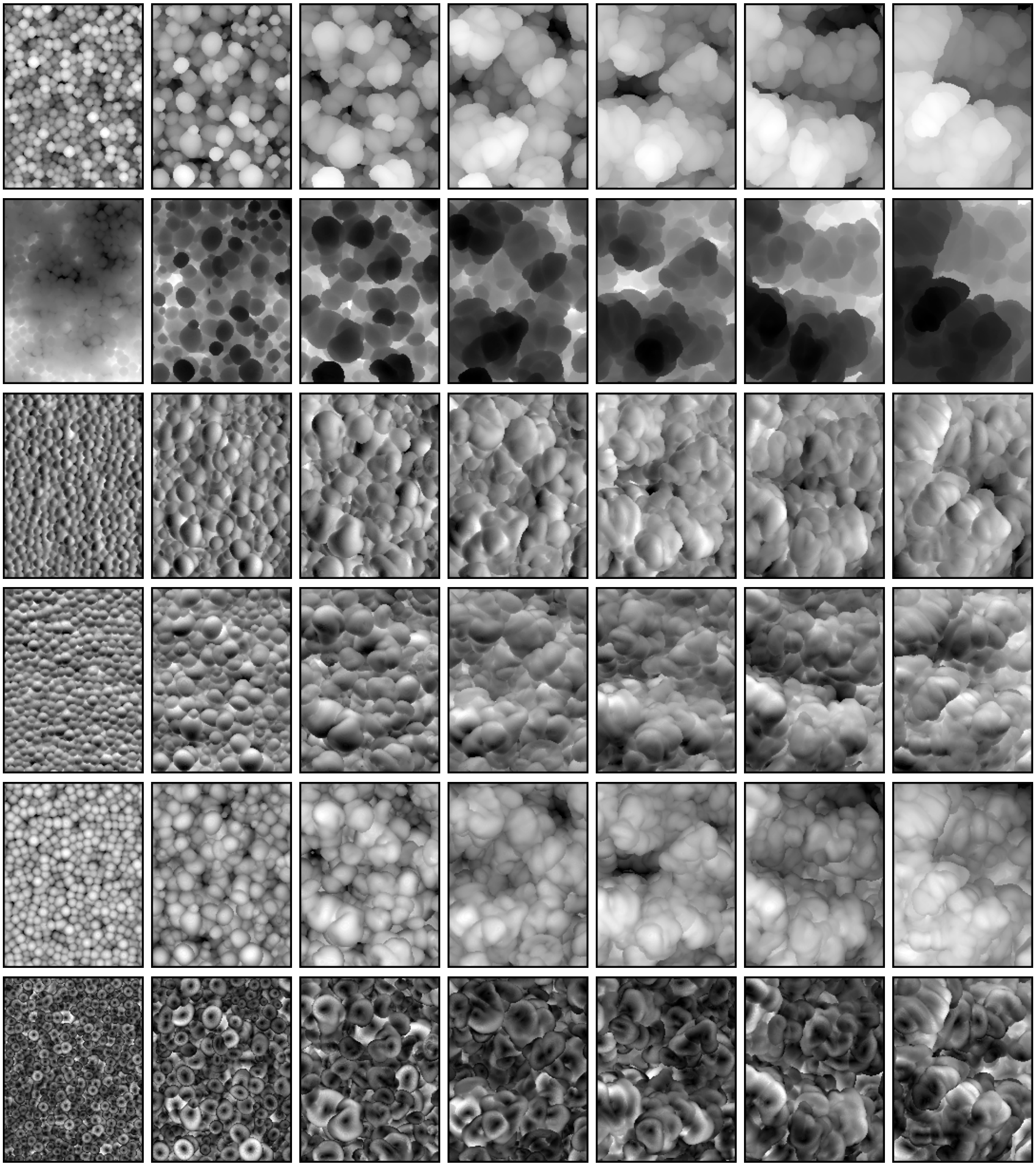


Figure 10: The other variables at the bubble boundary for the central 6 × 6 columns every 100-th time step, starting at time step 100, for the LES data. Each image is normalized so the intensity values lie between 0 and 255. The columns correspond to the time steps. The rows are, from top to bottom, the HDM, the pressure, x velocity, y velocity, z velocity, and the magnitude of the $x - y$ velocity.

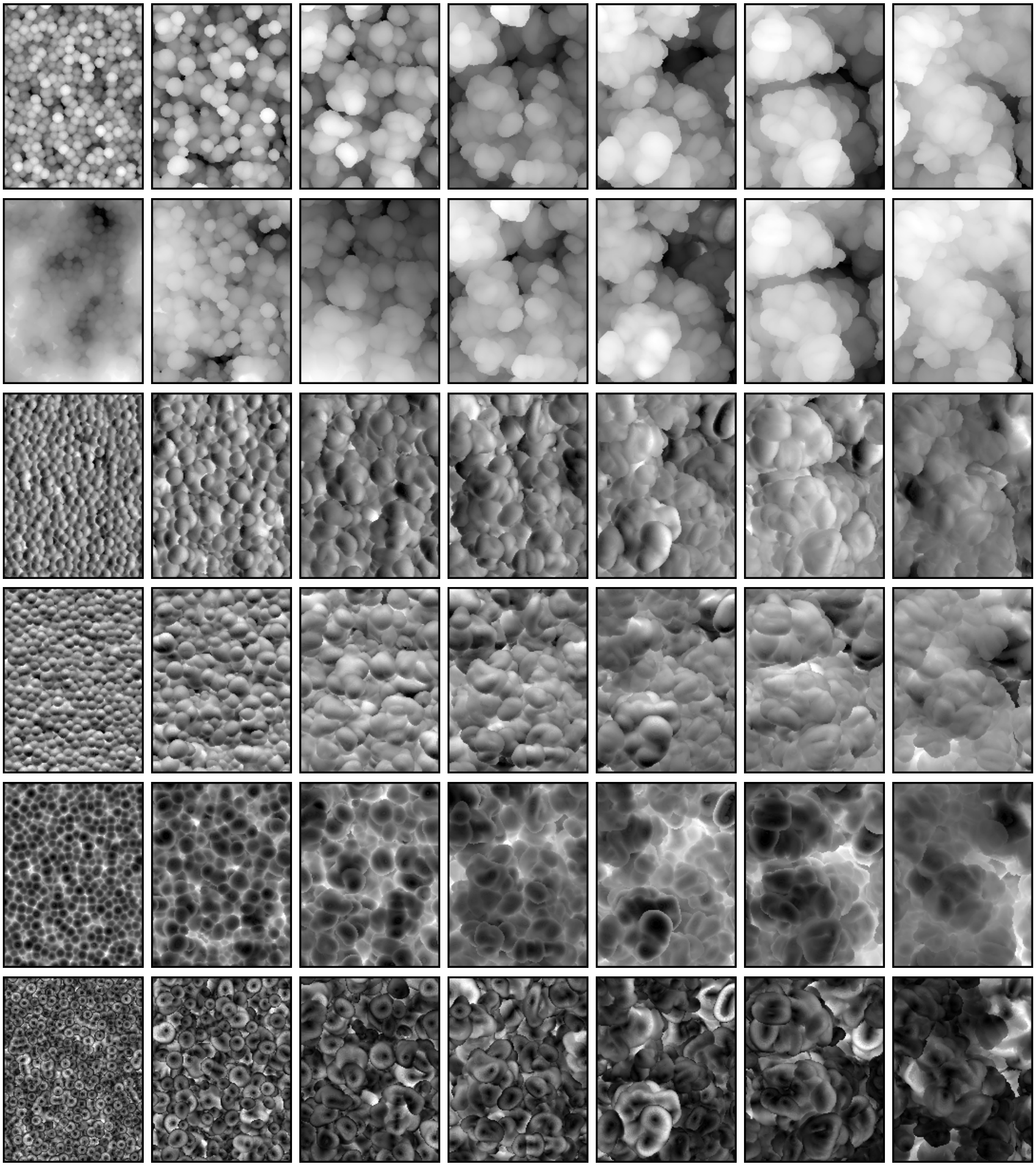


Figure 11: The other variables at the spike boundary for the central 6 × 6 columns every 100-th time step, starting at time step 100, for the LES data. Each image is normalized so the intensity values lie between 0 and 255. The columns correspond to the time steps. The rows are, from top to bottom, the HDM, the pressure, x velocity, y velocity, z velocity, and the magnitude of the $x - y$ velocity.

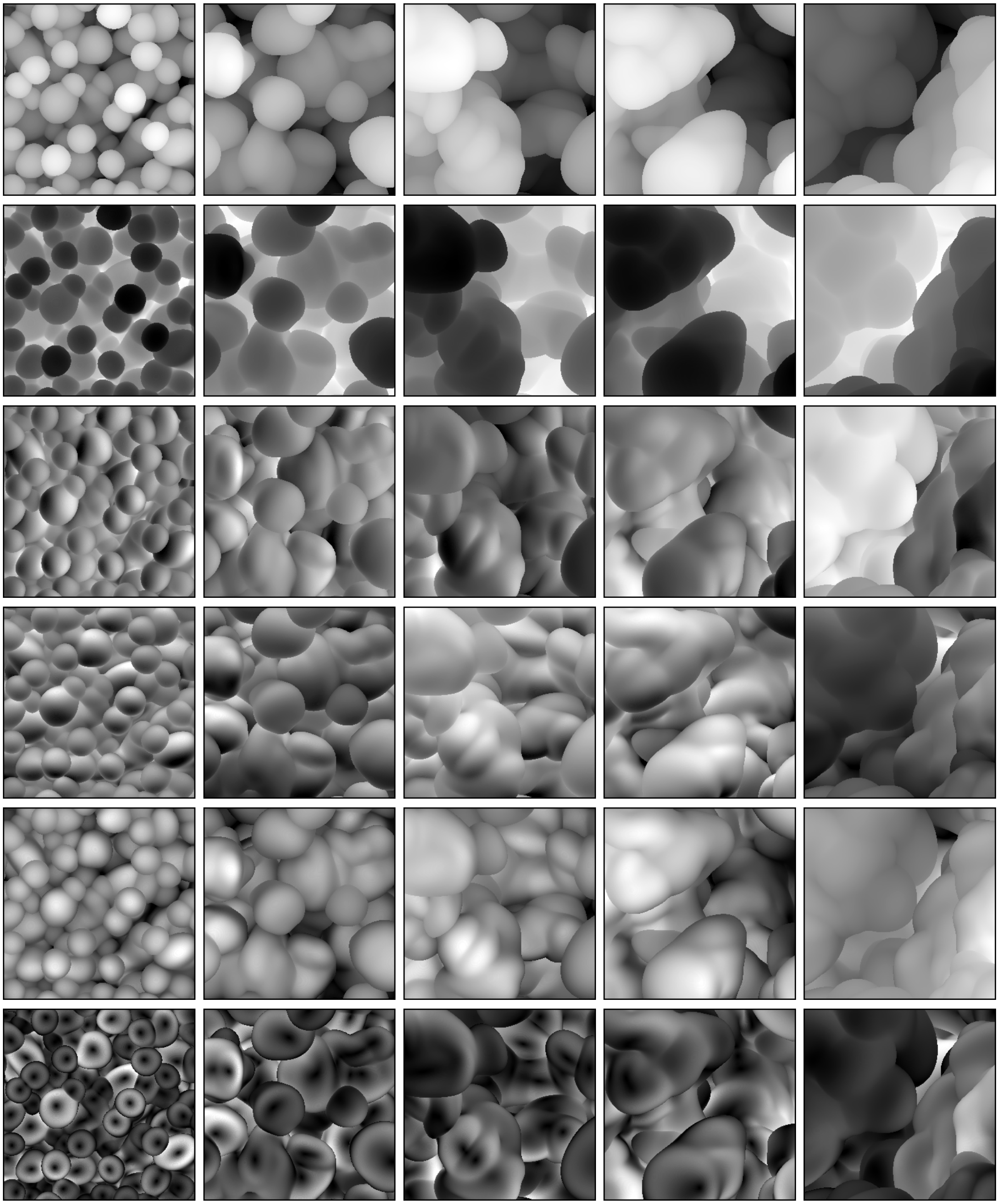


Figure 12: The other variables at the bubble boundary for the central 300 × 300 grid every 50-th time step, starting at time step 50 and ending at 247, for the DNS data. Each image is normalized so the intensity values lie between 0 and 255. The columns correspond to the time steps. The rows are, from top to bottom, the HDM, the pressure, x velocity, y velocity, z velocity, and the magnitude of the $x-y$ velocity.

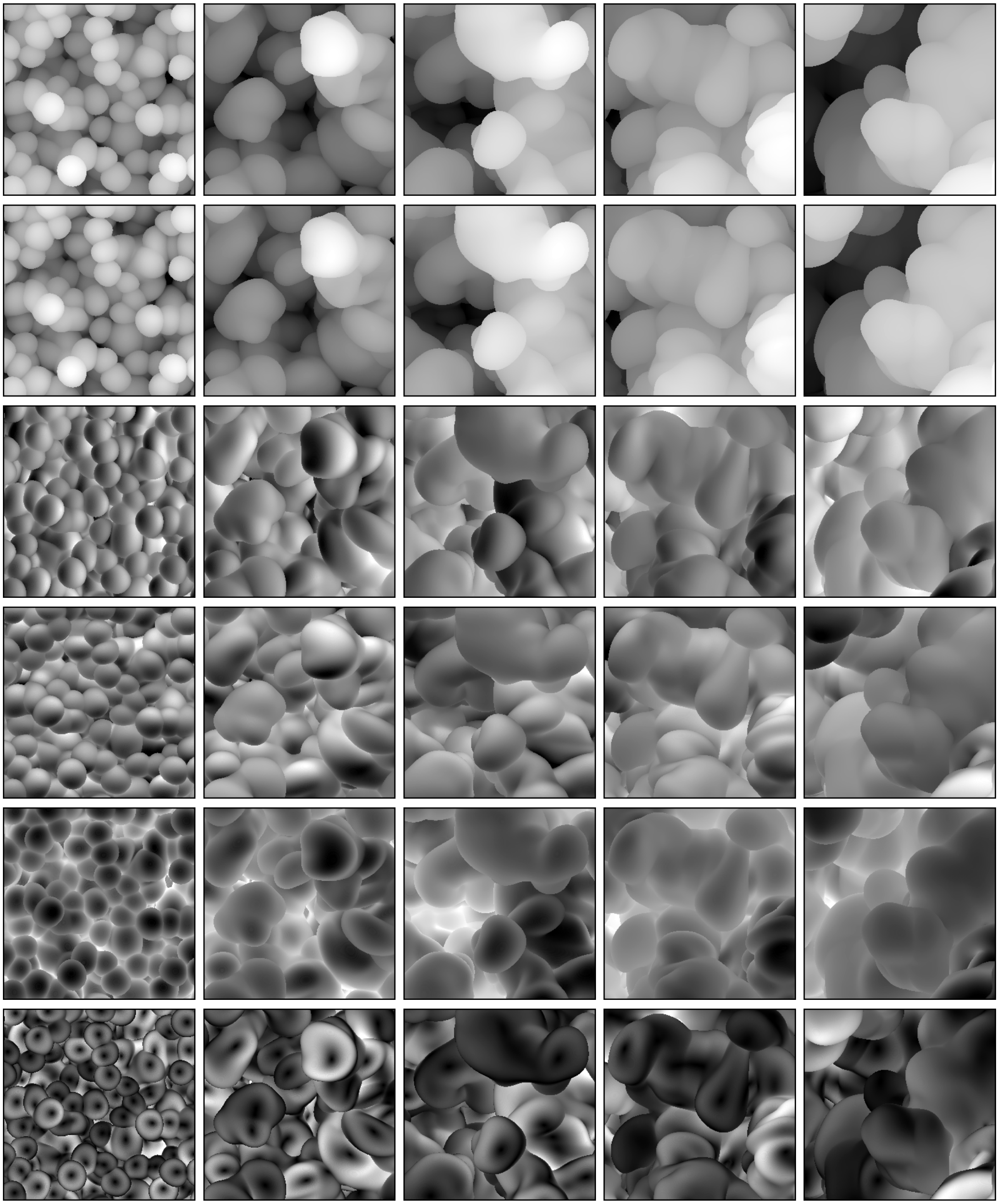


Figure 13: The other variables at the spike boundary for the central 300 × 300 grid every 50-th time step, starting at time step 50 and ending at 247, for the DNS data. Each image is normalized so the intensity values lie between 0 and 255. The columns correspond to the time steps. The rows are, from top to bottom, the HDM, the pressure, x velocity, y velocity, z velocity, and the magnitude of the $x - y$ velocity.

nique indicated that a single threshold was unlikely to work well for all time steps and we would need to vary the threshold with time. To determine the threshold values, we experimented with different thresholds for both the LES and the DNS data and visually inspected the resulting 2-D images to determine their quality. We examined how well the bubbles/spikes were defined, if their surface was continuous or had holes, if their 2-D boundary was smooth or jagged, and if the 2-D images of the different variables at the bubble/spike boundaries were clear or blurred. We next discuss our findings for the two datasets.

6.6.1 3-D thresholds for the LES data

Our experimentation indicated that for the LES data, the initial threshold for the bubbles should start at 2.75 and be gradually increased to 2.99 at time step 758. Similarly, for the spikes, the initial threshold should be 1.25 and the final threshold at time step 758 should be around 1.005. Once we obtained these initial and final values for the region growing thresholds, we had to determine a function to obtain the threshold over time. The simplest method we tried was linear interpolation given by Equations (1) and (2) for the bubbles and spikes, respectively. The plots of these threshold functions is shown in Figure 15.

$$T_b(t) = \frac{24}{75800}t + 2.75 \tag{1}$$

$$T_s(t) = -\frac{245}{758000}t + 1.25 \tag{2}$$

The results obtained from the linear interpolation however are less than adequate. One of the main problems is the deformation of the structure of the bubbles. For example, with the linearly defined thresholds, we could obtain bubbles with holes at the tip, bubbles which are disintegrated, or are completely missing, as shown in Figure 14(a). These deformations are caused by a low segmentation threshold. For the bubbles, reducing the threshold from its optimum range will result in the segmentation algorithm finding a shrunken bubble boundary which is internal to the bubble and in some cases, will find other structures far below the bubble boundary. A similar problem is observed when we select a spike threshold which is higher than the optimum range. This problem with the linear thresholding occurs predominately in the early- and mid-time steps and is gradually reduced at the later time steps.

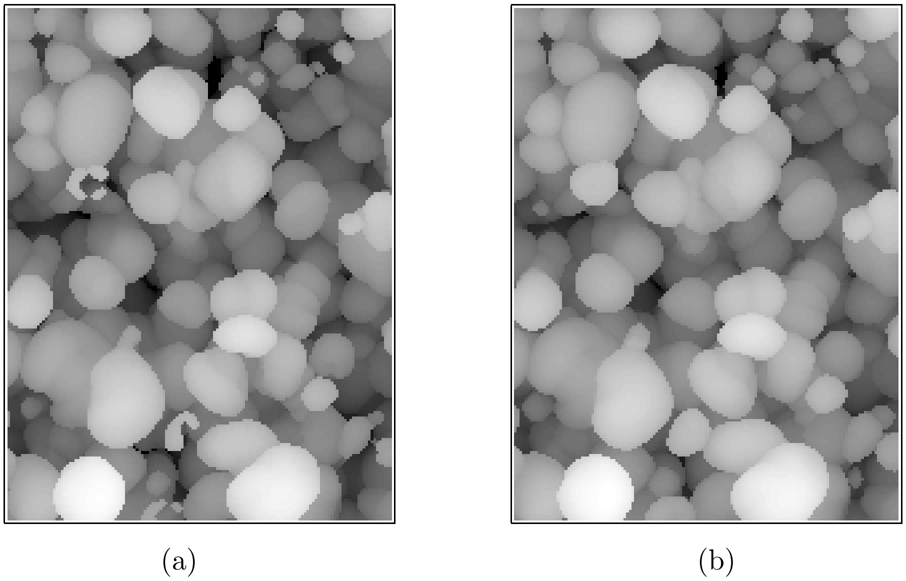
Based on this observation, and additional testing of threshold values, we concluded that a logarithmic threshold function provided good results for both the bubbles and the spikes. Equations (3) and (4) represent the logarithmic threshold functions for the bubbles and spikes, respectively.

$$T_b(t) = 2.75 + ln(1 + 0.04t)C_b \tag{3}$$

$$T_s(t) = 1.25 - ln(1 + 100t)C_s \tag{4}$$

where C_b and C_s are constants which satisfy the initial and final threshold conditions and are evaluated as follows with $t = 758$:

$$C_b = \frac{2.99 - 2.75}{ln(1 + 0.04t)} \tag{5}$$



(a) (b)

Figure 14: Height-depth map of bubbles from the 3-D region growing of the LES data at time step 250 of the central 6 × 6 columns. (a) HDM obtained using the linear threshold equation given in (1). (b) HDM obtained using the logarithmic equation given in (3). It is clear that the HDM from the linear threshold has deformations and some small bubbles that are completely missing.

$$C_s = \frac{1.25 - 1.005}{ln(1 + 100t)} \tag{6}$$

Figure 15 shows the logarithmic interpolation functions for the threshold for the LES data.

6.6.2 3-D thresholds for the DNS data

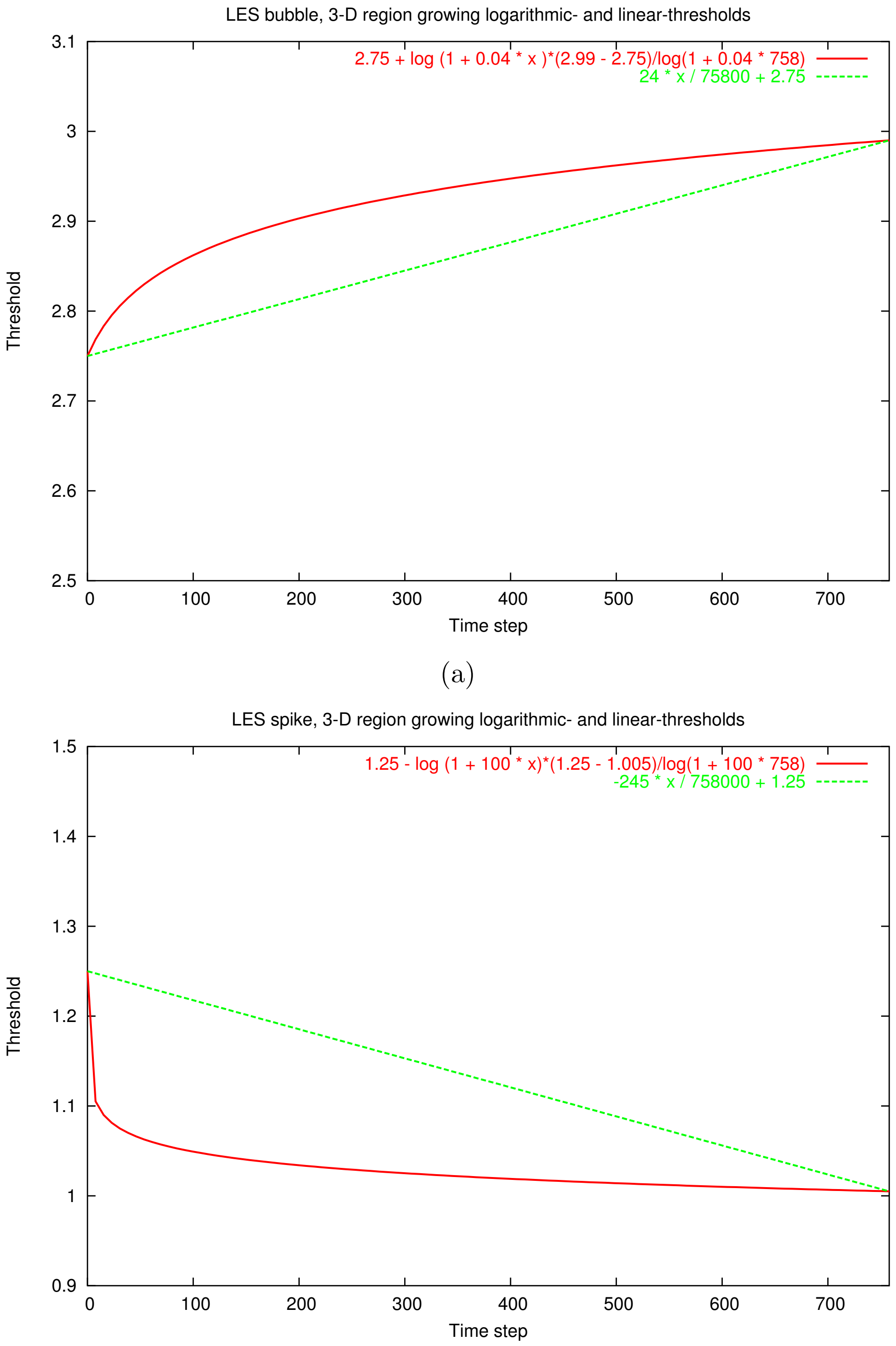
As with the LES data, our experiments indicated that a single threshold value does not work for all time steps for the DNS data. We also found that a linear threshold creates similar problems of deformation of bubbles as in the LES data. If we decrease (increase) the slope of the linear threshold to minimize this deformation for the bubbles (spikes) at one time step, it leads to jagged boundaries and the deformed bubbles now appear at a different time step.

Our experiments indicated that the initial threshold for the bubbles should start at 2.85 and gradually increase to 2.999 at time step 247. Similarly, the initial threshold for the spikes should start at 1.1 and gradually decrease to 1.0005 at time step 247.

To obtain good results for both the bubbles and spikes through out all the time steps, we used the logarithmic threshold provided in Equations (7) and (8) respectively.

$$T_b(t) = 2.85 + ln(1 + t)C_b \tag{7}$$

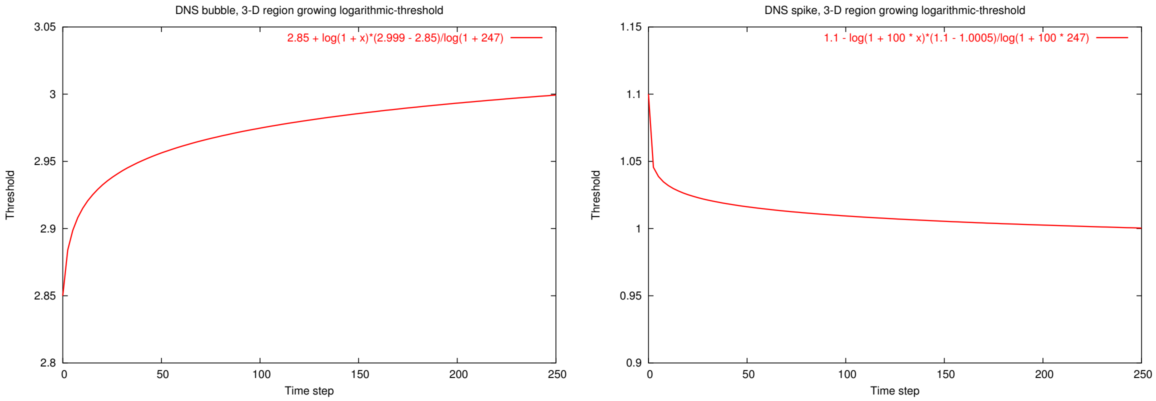
$$T_s(t) = 1.1 - ln(1 + 100t)C_s \tag{8}$$



(a)

(b)

Figure 15: Logarithmic- and linear-threshold plots for converting the 3-D LES data to 2-D using region-growing segmentation for (a) bubbles and (b) spikes. In green: the linear-threshold from equations (1) and (2). In red: the logarithmic threshold from equations (3) and (4).



(a) (b)

Figure 16: Logarithmic threshold plots for converting the 3-D DNS data to 2-D using region-growing segmentation for (a) bubbles and (b) spikes, using Equations (7) and (8).

where C_b and C_s are constants which satisfy the initial and final threshold conditions and are evaluated as follows with $t = 247$:

$$C_b = \frac{2.999 - 2.85}{ln(1+t)} \tag{9}$$

$$C_s = \frac{1.1 - 1.0005}{ln(1+100t)} \tag{10}$$

Figure 16 shows the corresponding plots of the threshold functions used for the DNS data.

6.6.3 Observations on the choice of 3-D thresholds

We next make some observations which are valid for both the LES and the DNS datasets. In the previous sections, we have described our use of the logarithmic threshold for converting the 3-D data to 2-D. We do not claim that the logarithmic functions used have any physical meaning for the Rayleigh-Taylor instability problem at hand; only that, in our experience, these functions provide a better threshold for obtaining the height-depth map images using our 3-D region growing algorithm.

Our main objective in using the logarithmic formulation was to minimize, if not eliminate, extensive deformation of the structure of the bubbles such as the ones shown in Figure 14. We also wanted to obtain height-depth map images where the bubbles and spikes are easy to identify, both visually as well as computationally. A logarithmic threshold enabled us to remove all the previously mentioned deformations throughout all the time steps. Our evaluation was done by a visual inspection of the height-depth map images as well as the suitability of the images of the other variables for subsequent processing.

At this point, it is important to note that the choice of thresholds can influence the different statistics for the bubbles in different ways. Our calculation of the thresholds was done to ensure that we could process the 2-D images to correctly obtain the number of bubbles and spikes over time. We have observed that the size of bubbles was far more sensitive to the choice of

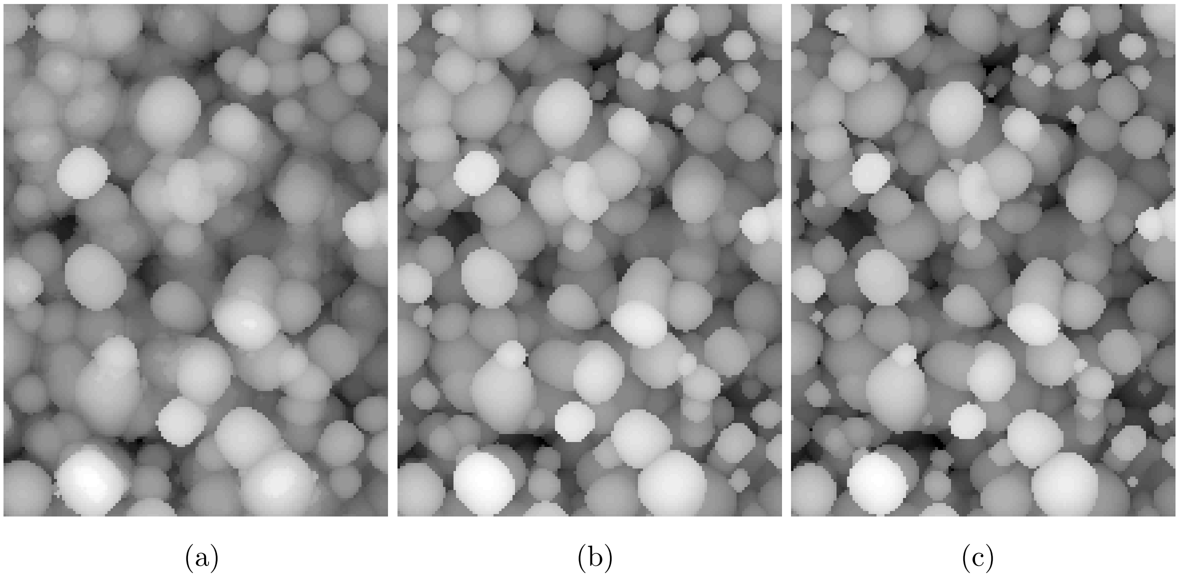


Figure 17: Size variation of bubbles due to threshold change at time step 200 of the central 6×6 columns of the LES data. The thresholds used to obtain (a) and (c) are 2.985 and 2.82, respectively. The threshold for (b) was obtained using the logarithmic equation in (3), which is roughly 2.903. We also see that some bubbles in (a), such as the bright one at the lower left corner, already have jagged boundaries.

thresholds, while the number of bubbles was relatively insensitive. This would make it easier to reliably obtain the number of bubbles rather than the size of the bubbles. As we can see from Figure 17, the bubbles are visually identifiable throughout the three examples while the size variation is dramatic. However, we note that as the bubble size increases beyond a certain point, it becomes harder to segment the image to obtain the 2-D boundary of the bubbles due to the blurring effects at the boundary of each bubble. Therefore, we cannot expect to obtain the size of the bubbles precisely; we can however, obtain the distribution of the sizes of the bubbles.

As we shall see next, our choice of the logarithmic threshold makes subsequent processing easier as it results in clear images for all the variables at the bubble boundaries, enabling us to use these variables in the process of bubble counting. However, we also observe that, as with any technique based on thresholding, we need to address the question of the sensitivity of our results to the choice of thresholds. Our selection of the region growing method, instead of the Canny edge detector, implies that we only need to study the sensitivity of the results to a single threshold, a topic we discuss further in Section 9.

7 Analysis Part II: Counting Bubbles/Spikes in the 2-D Data

In this report, we focus on extracting the bubble and spike counts for the LES and the DNS datasets. We will discuss the extraction of other statistics, such as the bubble sizes, the number of neighbors, and the average distances between bubbles in a follow-on report.

We extract the bubble/spike count by first using traditional image processing techniques to segment the 2-D images in Figures 10 through 13 and then counting each individual bubble. Though these 2-D images are much smaller than the original 3-D data, we once again rejected the more complex techniques such as active contours as they require the setting of several parameters, and being iterative in nature, are rather time consuming. A possibility was to exploit the circular structure of the bubbles and use template matching, but this approach would not have been applicable at later time, when the bubbles are no longer circular.

We considered several segmentation techniques to identify the individual bubbles. These included the watershed algorithm, the Canny edge detector, and region growing techniques applied to the different variables at the bubble boundary [17]. As we have access to more than one variable at the bubble boundary, we can also exploit the additional information to improve the quality of our analysis. We also realized that for counting the bubbles, we did not need to identify the boundary of each bubble; instead, we could focus on a characteristic region associated with each bubble, such as its tip. We next describe the different techniques we used for counting the bubbles and spikes in the LES and DNS datasets.

7.1 Region growing methods

Our approach to the use of region-growing segmentation for counting the bubbles was motivated by the observation that in the height-depth map image, the region at the tip of a bubble contained the highest point of a bubble and the height of the surrounding region was within some small threshold of this peak. This is especially true at the early- to mid-time steps, when the bubble structures are well-defined. This tip region may not encompass the entire bubble, but, as we observed earlier, we do not need to obtain the entire region for each bubble.

Our region growing algorithm follows the traditional description [17], where the image is initially subdivided into small homogeneous regions which are then allowed to grow by merging with neighboring connected regions. In our case, we use each pixel from the height-depth map image as an initial region. Then, a single region is selected and all neighboring regions are tested for homogeneity with the current region. If the two regions are homogeneous, using some homogeneity metric, then they are merged, and the process continues.

There are three issues which affect the segmentation obtained using a region growing technique:

- **Homogeneity Metric**

  We have used two homogeneity metrics for our region growing algorithm. These are based on our earlier observation that pixels in the tip region of a bubble are similar in their height measured off the initial interface between the two fluids. The first metric is based on the mean of a region, where a pixel is merged into an adjacent region if the height of the pixel is within a threshold of the mean of the region. The second metric is based on the variance of a region, where a pixel is merged into an adjacent region if the variance in height of the combined region is below a threshold. These metrics are referred to as the mean homogeneity metric and the variance homogeneity metric, respectively. The image is segmented by merging adjacent regions as long as the new region is also homogeneous according to the metric used. Figures 18 and 19 show the results using the mean and variance homogeneity metrics, respectively.

  Note that it is possible to make a second pass through the image and merge adjacent regions found during the first pass, provided they meet the homogeneity metric. We did not incorporate this option in our segmentation.

- **Pixel ordering**

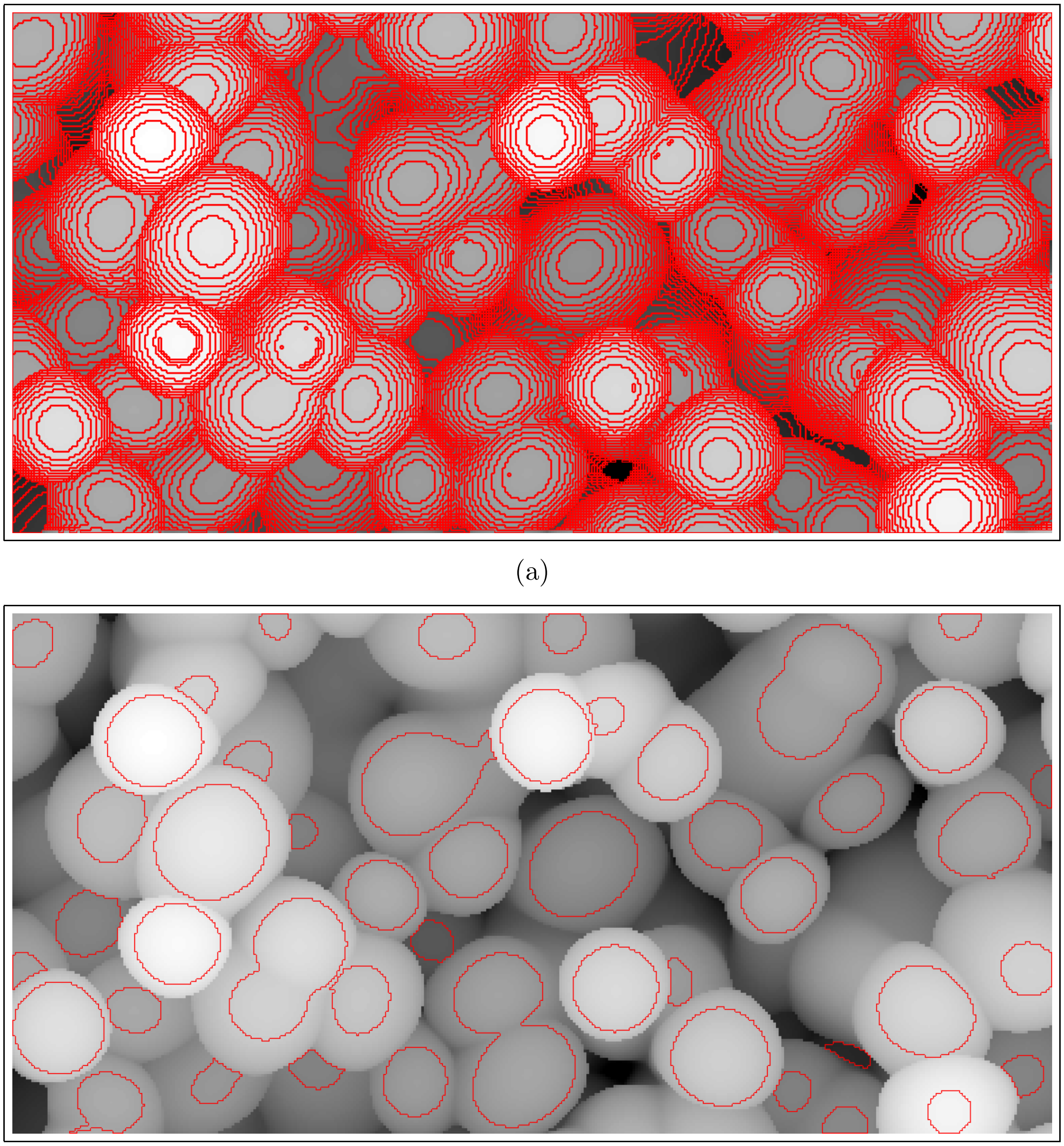
  It is well known that the order in which the regions are processed determines the quality of the final segmentation. This is true in our problem as well. As we are interested in the tips of the bubbles which are regions with higher height values than the other parts of the bubble, we first sort the initial regions (pixels) in descending order. We then process each of the highest pixels and merge the adjacent regions to them. Subsequent brightest regions are then processed assuming these regions have not already been merged to other regions. This ordering exploits our knowledge of the data and provides better results than does a linear processing of the regions.

- **Region cleanup**

  Once the segmentation was performed, we found that the image was oversegmented and required cleanup to retain only the tip regions of the bubbles. Ideally, we would like a 1-1 correspondence between every region and a bubble. In other words, we want to minimize the cases where multiple regions are used to identify a bubble or there are bubbles without a region representing them. This implies that the cleanup process should remove those regions obtained during segmentation which do not correspond to the tip of a bubble. We accomplish this by exploiting properties of the regions around the tip of the bubble and regions around the boundary of the bubble.

  In most cases, we observe that the regions around the tip of the bubble are larger in size than the regions surrounding the tip and the regions around the boundary of the bubbles. This is visually clear in Figures 18 (a) and 19 (a), which correspond to sample images segmented using the mean and the variance homogeneity metrics, respectively. The segmented regions in these images are indicated by a red boundary. We also see that, compared to the central region of a bubble, the area around the boundary of the bubbles is over segmented (indicated by the higher concentration of red pixels), with several small regions. We use this information to remove the extra regions around the boundary of bubbles based on a size threshold which is fixed for all time steps. In addition, we observed some of these extra regions are elongated or oddly shaped unlike the bubble tips especially at the early time steps (50 through 300 for LES and 20 through 150 for DNS). We exploit this information and constrain our bubble tips to have a large aspect ratio and a high ratio between the number of pixels within a region and the size of the smallest encompassing rectangle. As mentioned earlier, bubble tips are usually round and higher than the surrounding region; so, we require the center of mass of a bubble tip to belong within the region itself and be at least a few pixels above the initial fluid interface. All regions that are contained entirely in another region are also merged into the encompassing region as long as the new region is a bubble tip according to the above restrictions. Figures 18(b) and 19(b) show the results of region growing segmentation after this cleanup for the mean- and variance-homogeneity metrics, respectively.

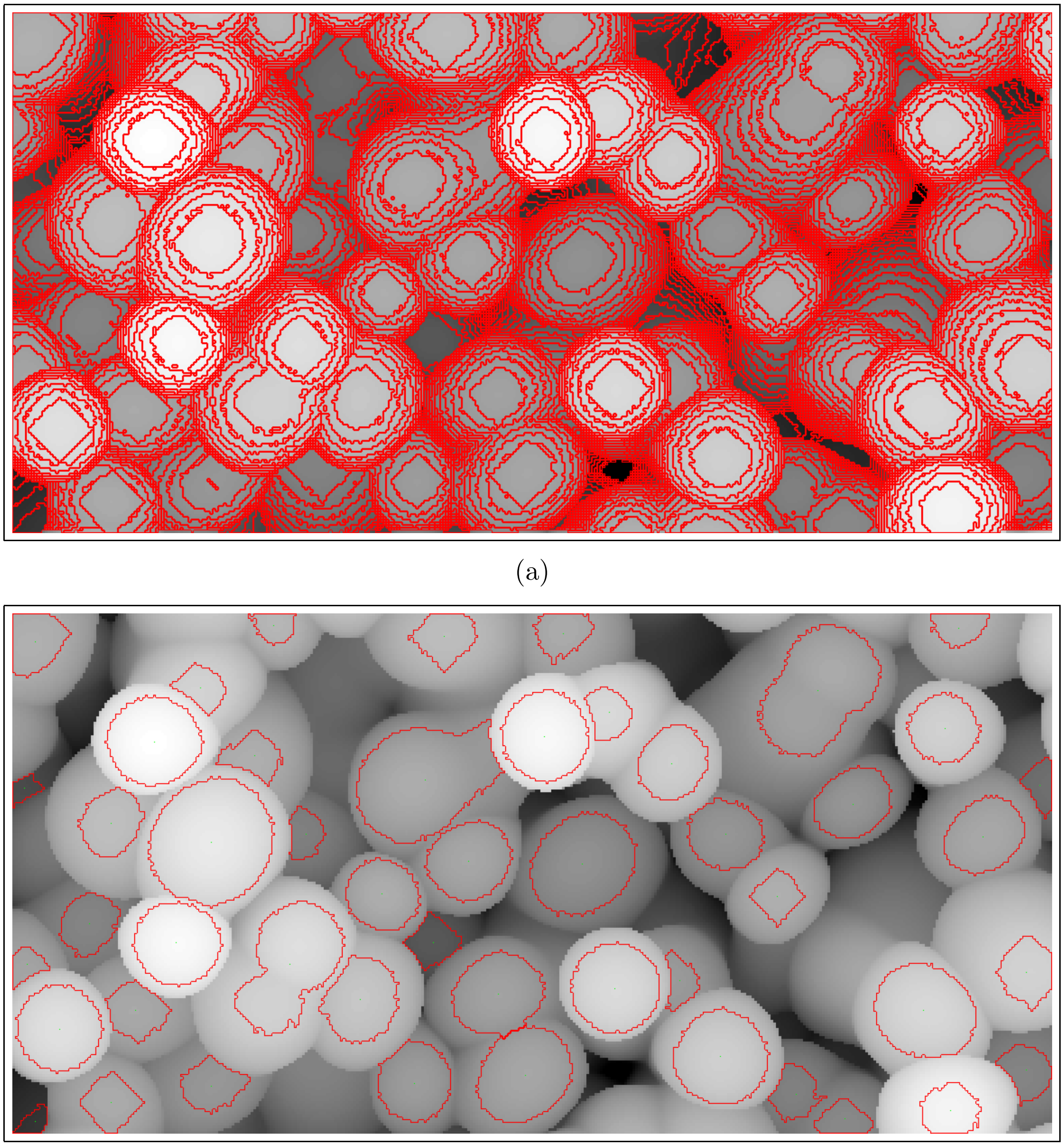
  Unfortunately, this cleanup process will usually remove a small number of bubble tips as they meet one or more of these removal criteria. These bubbles primarily have their tip partially or entirely covered by another bubble. The remaining visible portion of these



(a)

(b)

Figure 18: Region-growing segmentation of the bubble height-depth map image for a sub-image at time step 50 from the DNS data. (a) The regions are obtained using a mean homogeneity metric. The boundary of each region is indicated in red. (b) Results after the cleanup process is applied to (a).



(a)

(b)

Figure 19: Region-growing segmentation of the bubble height-depth map image for a sub-image at time step 50 from the DNS data. (a) The regions are obtained using a variance homogeneity metric. The boundary of each region is indicated in red. (b) Results after the cleanup process is applied to (a).

bubbles is represented by regions which are small, elongated or arched. They are removed as they meet one or more of the removal criteria. We believe this is not a problem because these bubbles will be entirely covered by, or be incorporated into, other bubbles at later time steps. The only side effect of this is that a bubble stops being counted when its tip region is not seen in the height-depth map image, though parts of the bubble are clearly visible.

Figures 18 and 19 indicate that the results from the mean and the variance-based homogeneity metrics are very similar. However, the regions created by the variance metric have boundaries which are more jagged than those created by the mean metric.

7.2 Method based on the magnitude of the $x - y$ velocity

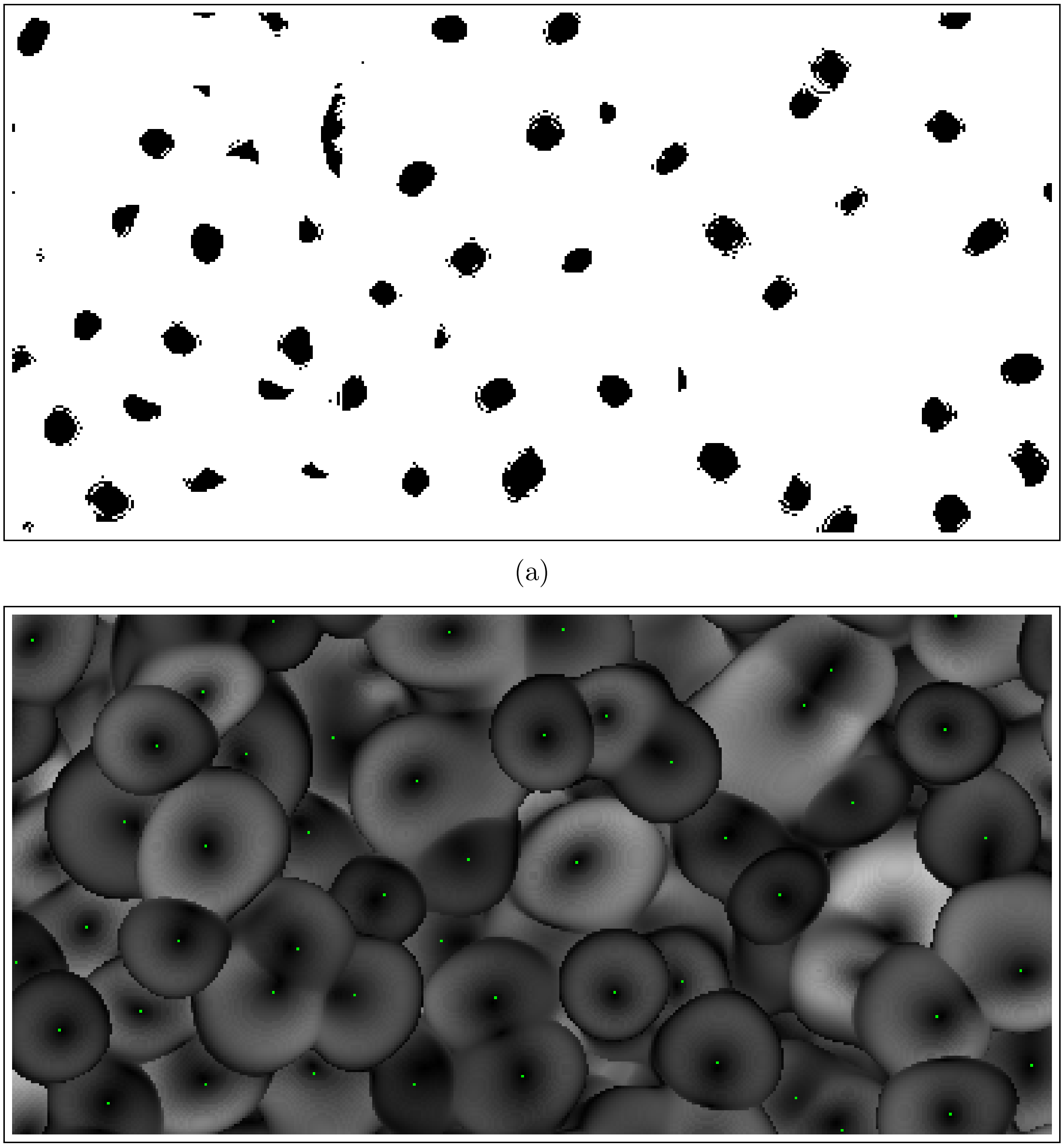
The second approach we considered for computing the number of bubbles is based on the magnitude of the $x - y$ velocity. The x and y velocities at the bubble surface individually do not provide a simple method for finding the bubble tip regions. However, one property of the x velocity is that in many cases, a bubble has positive x velocity on the right side and negative x velocity on the left side. Similarly, the y velocity is usually positive in the top half of the bubble and negative otherwise. By combining these two images to obtain an image with the magnitude of the $x - y$ velocity, we see that the bubble tips have a small $x - y$ velocity magnitude (indicated as the darker regions at the centers of the bubbles, especially at early time). This can be seen clearly in the last row of the images in Figures 10 through 13. These images also indicate that the boundary of the bubbles have a small $x - y$ velocity magnitude.

We differentiate these boundary regions from the central bubble region by using the height-depth map images. At the bubble boundary, there is a larger height variance than around the tip of the bubble. Thus, for each pixel identified as having an $x - y$ velocity magnitude less than a threshold, we consider a small window centered at the pixel in the height-depth map image. If the height variation within the window is small, the pixel is considered to be in a region around the tip of the bubble; otherwise the pixel is considered to be at the boundary of the bubble. Once all pixels with a small $x - y$ velocity magnitude satisfying the height variation constraint are identified, all connected pixels are considered as a single region at the tip of the bubble. In some cases, there are a small number of pixels, around ten or so, which form a connected region within a bubble which already has a larger connected region around its tip. If left as is, these connected pixels would result in the identification of multiple tips for a single bubble. We eliminate these extra regions by using a size threshold. Alternatively, we could have used morphological operations to incorporate the smaller regions into the larger ones. Figure 20(a) shows the connected pixels at the bubble tip and in panel (b), the center of mass of each region overlaid on the magnitude of $x - y$ velocity image.

We investigated a number of methods to separate the pixels with a small magnitude of $x - y$ velocity at the tip of a bubble from those at the boundary of a bubble. These include:

- The variance approach which evaluates the height variance within a small window centered at the pixel in question.

- The mean approach which computes the mean of the window centered at the pixel and compares it to each pixel within a slightly larger window centered at that pixel.



(a)

(b)

Figure 20: The black pixels in (a) represent the pixels with a small magnitude of $x-y$ velocity which are around the tip of bubbles. The small dark regions around the larger dark regions have not yet been removed to indicate their locations. In (b), the center of mass for the regions in (a) is displayed (in green) on the magnitude of $x-y$ velocity image after the cleanup step to remove the small isolated black regions. This is a sub-image from the DNS data at time step 50.

- The center approach which simply checks if the height of the center pixel is within a threshold of every pixel in the window.
- The variance $z-$velocity method evaluates the height variance within a small window centered at the pixel in question and, in addition, requires that the $z-$velocity of the pixel be larger than a small negative number.

In addition, pixels whose height from the initial fluid interface was less than a small threshold are considered as background pixels and are not included in the bubbles/spikes.

In these methods, we used the same thresholds for all time steps, though the thresholds were different for the LES and DNS datasets. The first three methods of separating bubble tip regions from bubble boundary regions provided good results for both the LES and DNS data set. However, we observed that at early time steps (before time step 40 for LES and time step 20 for DNS), all three methods identify more of the bubble boundary pixels as bubble tip pixels because the height variation is much smaller at those time steps. This is a side effect of keeping the thresholds fixed for all time steps. To address this, we considered the fourth method, which exploits the fact that at early time, the $z-$ velocity is negative at the boundary of bubbles and positive at the tip of the bubbles. However, we expect this method will result in lower number of bubbles at later time steps as there are bubbles with entirely negative $z-$velocity which will not be counted.

An alternative approach to using the magnitude of the $x-y$ velocity is to consider a simple implementation of our original observation. If we highlight all pixels which have the $x-$velocity of its right neighbor opposite in sign to the $x-$velocity of its left neighbor, we get a vertical line centered along the bubble. Similarly, if we highlight all pixels which have the $y-$velocity of its top neighbor opposite in sign to the $y-$velocity of its bottom neighbor, we get a horizontal line centered along the bubble. The intersection of these two lines indicates a bubble tip. We named this approach the 'hot-cross buns' algorithm for obvious reasons. While it does not depend on any parameters, it can miss bubbles where either the $x-$ or the $y-$velocity does not change sign at the bubble boundary. Further, in a manner similar to the approach based on the magnitude of the $x-y$ velocity, it can count bubble tips at the 2-D boundaries of bubbles. These can be resolved as in the earlier case by considering the height-depth map image.

7.3 Bubble/Spike counts at early and late time

The methods we used for counting the bubbles and spikes were motivated mainly by what we saw in the images of the height-depth map and the various variables at the bubble boundary. Our early focus was on the middle time steps ranging from 50 to 300 for the LES data and from 20 to 150 for the DNS data. In this range, we avoid the issues with bubble definition at the early time steps, where bubbles are still forming from the initial perturbation, or the issues at late time steps, where bubbles have merged to form complex structures and it is unclear how best to define a bubble based on the 2-D images. One option to address this problem of bubble definition is to use different algorithms for the different time ranges. However, we prefer to use a single algorithm over the entire dataset as this would avoid the issues associated with interpreting the results when we switch from one algorithm to another over the course of the dataset.

In this report, we assume that the way in which we have defined the bubbles and spikes in our algorithms is valid for the entire dataset, that is, the same definition is valid at early time, mid-time, and late time. We will revisit this issue in the follow-on report, where we will consider bubble/spike counts derived using algorithms which focus on the size of the structures.

7.4 Observations on bubble/spike counting algorithms

The two main approaches for bubble counting have their pros and cons. The region growing approaches, whether based on the mean- or the variance homogeneity metrics, are non-trivial to implement. The region-growing phase requires the data structures to keep track of the regions as they grow from a single initial pixel, while the cleanup phase requires the identification of regions which satisfy several constraints simultaneously. Each of these constraints requires the setting of a threshold, which must be chosen appropriately to give correct results over a large range of time steps. These techniques also require additional memory to keep track of the regions created during segmentation. They are also overall more computationally expensive as they require an initial sorting, followed by the region-growing, and then cleanup.

On the other hand, the magnitude $x - y$ velocity based methods are relatively simple to implement. They also exploit other variables in addition to the height-depth maps and therefore are likely to be more robust. They have fewer thresholds than the region-growing methods and some additional memory is required for the other variables used in the computation.

In Sections 8.1 and 8.2 we will compare these methods based on a visual evaluation of the bubbles identified as well as the counts of the bubbles and spikes over time for the LES and the DNS datasets.

8 Analysis Results

We next present our results for the bubble and spike counts for the LES and DNS datasets using the different methods for bubble counting described in Section 7. The parameters used for the different methods are as follows:

- Region growing methods

  The thresholds used for the mean homogeneity metric for the LES dataset starts at 1.01 at time step 50 and increases linearly to 1.8 at time step 758. For the variance metric for the LES dataset, the threshold starts at 0.05 and increases linearly to 0.7. For the DNS dataset, we can keep the mean and the variance homogeneity metric thresholds constant at 1.2 and 0.8, respectively, for all time steps. Recall that these techniques are applied to the height-depth map images and used to divide the entire image into homogeneous regions. The higher resolution of the DNS data allows us to use the same threshold for all time steps.

  The thresholds used in the cleanup are independent of the metric used to generate the segmentation. For the LES (DNS) data, regions smaller than 10 (81) pixels are considered small and removed; regions with aspect ratio smaller than 0.5 (0.5) are removed; and regions where the ratio of the number of pixels to the area of the encompassing rectangle is less than 0.5 (0.5) are removed. Note that, in addition, we remove regions whose center of mass lies ouside the region and merge regions where one lies entirely inside another.

Further, pixels whose height is less than or equal to 2 (5) grid points are considered as background pixels for the LES (DNS) datasets and are not included in the bubbles/spikes.

- Methods based on the magnitude of the $x - y$ velocity

  The variants of this method all start by identifying pixels with the magnitude of the $x-y$ velocity smaller than a threshold. This threshold is set at 0.3 (0.49 for bubbles and 0.64 for spikes) for the LES (DNS) dataset. The methods differ in the way we differentiate between pixels with small magnitude of $x - y$ velocity which are at the tip of a bubble or at the boundary. The variance approach uses a threshold of 0.6 (0.5) for the LES (DNS) data, while the mean approach uses a threshold of 2 (2) pixels and the center approach uses a threshold of 2 (2) pixels. The variance $z-$velocity method uses the same threshold as the variance method and keeps only those pixels whose $z-$velocity is greater than -0.15 (-0.15). Each of these uses a window of size 5×5 (7×7) to differentiate between pixels at the bubble tip and pixels at the bubble boundary. The smaller window used in the mean approach to obtain the mean height of the region is 3×3 (5×5) for the LES (DNS) dataset.

  The size threshold used for removing small regions near the bubble tips is 2 (14 for bubbles and 13 for spikes) for the LES (DNS) data. In addition, pixels whose height is less than or equal to 2 (5) pixels are considered as background pixels for the LES (DNS) datasets and are not included in the bubbles/spikes.

For our work, except for the thresholds in the homogeneity metrics for the LES data, we used constant threshold values for all time steps. These thresholds were selected by visually examining the bubble tips identified by each method at a few time steps which were equidistant in log time and spanned the full simulation. As we have mentioned earlier, given the scale of the structures of interest, it is unclear if constant thresholds are an appropriate choice, especially at the very early and very late time steps. However, we found that a constant threshold worked well over a wide range of time steps spanning the middle of the simulation. As we shall show later, there are issues with the bubble counts at the very early and late time. In both these situations, it is unclear what exactly is a bubble, a question we must address first before we can count the bubbles at these time steps.

There are several aspects to the results of our analysis, ranging from the visual comparison of different methods for counting bubbles, to the visual identification of the bubbles at different time steps, and the plots of the bubble counts. In the following, the results visually illustrating the counts will be presented only for the bubbles, as the results for spikes are similar.

8.1 Visual inspection of bubble counts

First, we present sample images to illustrate the results obtained by the different bubble counting methods. For the LES data, Figure 21 shows the bubble height-depth map images at time step 100 with the bubble tip highlighted in red for the methods based on the magnitude of the $x - y$ velocity and the region-growing segmentation. The corresponding images for the DNS data, at time step 50 are given in Figure 22. For the DNS data, we indicate the bubble tips by a 5×5 red square as a single pixel is difficult to see in these higher resolution images.

Note that these images indicate the bubble tips on the height-depth map which has been normalized to have values between 0 and 255. Care must be taken to interpret these images as what appears to be a "large" gradient between "two" different bubbles may be just an effect of the normalization.

Next, using the method based on the magnitude of the $x-y$ velocity and the variance approach to differentiate between pixels at the bubble tip and the bubble boundary, we show images of the central 6×6 columns of the LES data at every 100-th time step in Figure 23. The bubble tip is superimposed in red. Figure 24 shows similar results for the DNS data for the central 300×300 pixel columns at every 50-th time step. For the DNS data, as in the earlier images, we indicate the bubble tips by a 5×5 red square as a single pixel is difficult to see in these higher resolution images.

Figures 21 and 22 indicate that the different methods for counting bubbles are very similar at least at time step 100 for the LES data and time step 50 for the DNS data. We observe that in the LES data, the center and the mean approach based on the magnitude of the $x-y$ velocity tend to count bubble tips at the boundary of two bubbles. This is less pronounced in the DNS data, given its higher resolution. In both datasets, the variance $z-$velocity approach misses those bubbles with a negative $z-$velocity, as was expected. In the LES data, the region-growing approach based on the variance metric tends to miss fewer bubble tips. The two region-growing methods perform similarly for the DNS dataset.

The results in Figures 23 and 24 for the identification of the bubble tips using the variance-based magnitude of the $x-y$ velocity method indicate that for the LES method, we do well until time step 300, after which we appear to overcount the bubbles tips as indicated by tips which are relatively close to each other. This is a side effect of keeping the size of the window over which the variance is obtained fixed for all time steps. As the bubbles are larger at later time, the overcounting will be reduced if we increase the size of the window. However, we first need to address the issue of a consistent definition of a bubble at later time before we fine tune the parameters of our algorithms.

As illustrative examples, Figures 25 and 26 show the bubble tips identified on a large sub-image at time step 200 and time step 50 for the LES and DNS data, respectively.

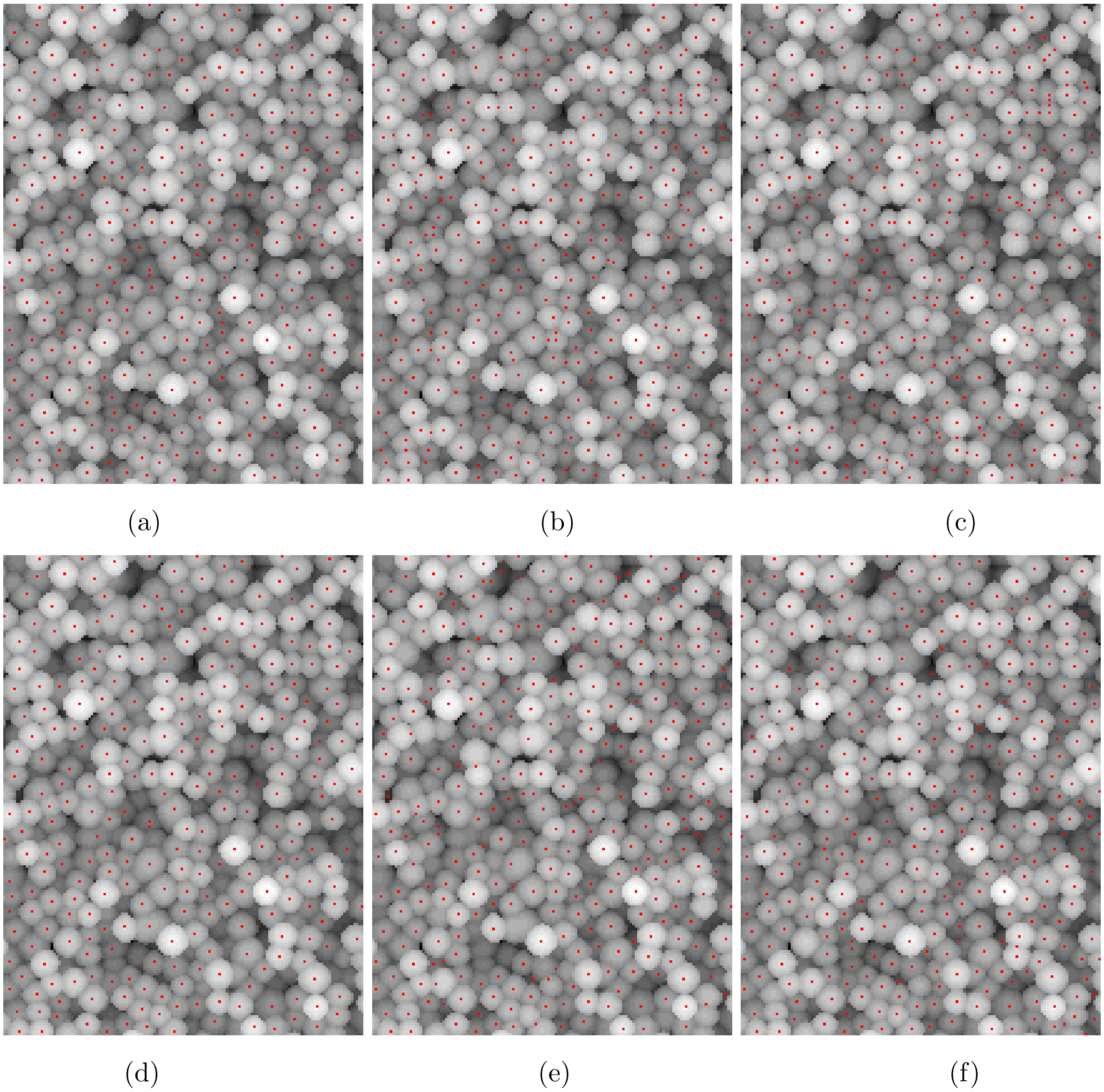
8.2 Bubble- and spike-count plots for LES and DNS data

We next discuss the bubble and spike counts obtained by applying our counting algorithms to the 2-D images. As mentioned earlier, in this report, we assume that a single algorithm, with constant thresholds (unless mentioned otherwise), will yield corrent counts over all time steps. We know that this is not quite correct at the very early time steps, where the bubbles are still being formed, and at the very late time steps, where the bubble definition is unclear, or the scale of the structures such that constant thresholds will lead to over-counting. In the follow-on report, we plan to revisit this issue of bubble definition at later time in the context of generating statistics on bubble sizes.

Figures 27 and 28 are the plots of the bubble and spike counts over time for the LES data, respectively. The corresponding plots for the DNS data are given in Figures 29 and 30. These plots are on a log-log scale.

From an algorithm viewpoint, these plots indicate the following:

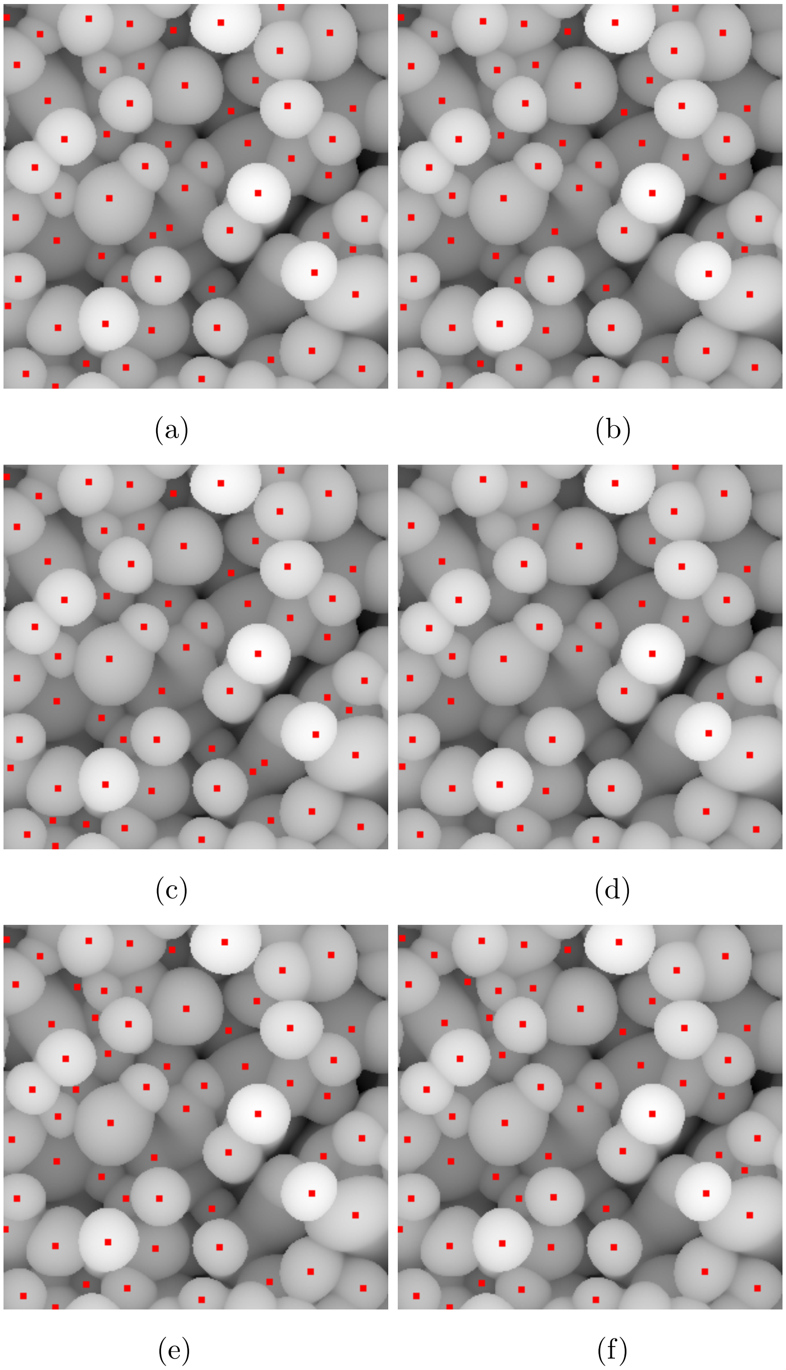
- For the DNS data, the counts obtained from the different bubble/spike counting methods are very close for both the bubbles and spikes. The only exception is the method based on the magnitude of the $x-y$ velocity, which along with the variance, uses the $z-$ velocity



(a) (b) (c)

(d) (e) (f)

Figure 21: Counting the bubbles in the LES data, center 6 × 6 columns, time step 100. The first four images are obtained using the method based on the magnitude of the $x - y$ velocity, with the height variance obtained using (a) the variance approach, (b) the mean approach, (c) the center approach, and (d) the variance $z-$velocity method. The last two images are obtained using the method based on the region-growing segmentation with (e) the mean homogeneity metric and (f) the variance homogeneity metric.



(a) (b) (c) (d) (e) (f)

Figure 22: Counting the bubbles in the DNS data, center 300 × 300 pixel columns, time step 50. The first four images use the method based on the magnitude of the $x-y$ velocity with the height variance obtained using (a) the variance approach, (b) the mean approach, (c) the center approach, and (d) the variance $z-$velocity method. The last two images are obtained using the method based on the region-growing segmentation with (e) the mean homogeneity metric and (f) the variance homogeneity metric.

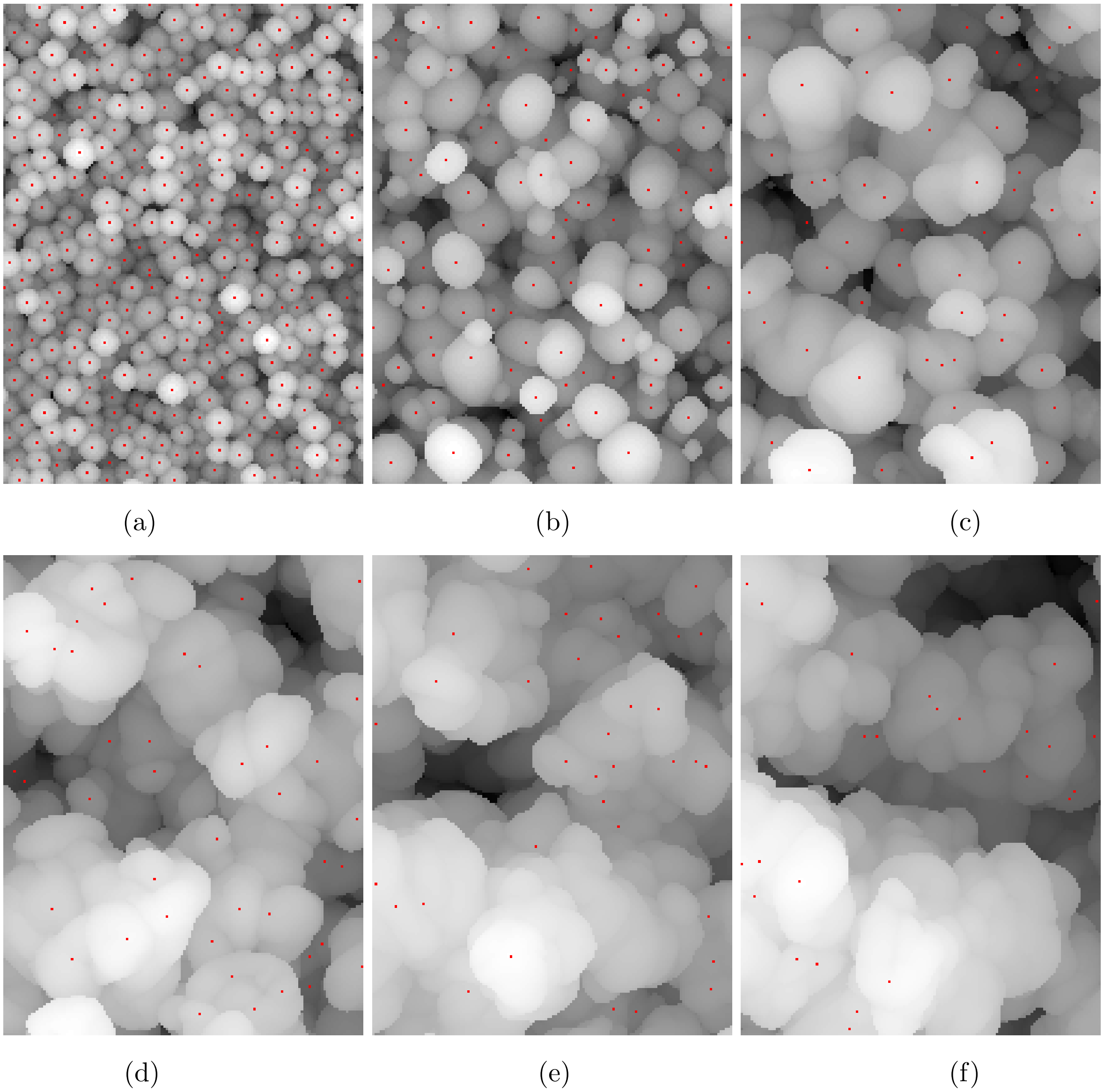
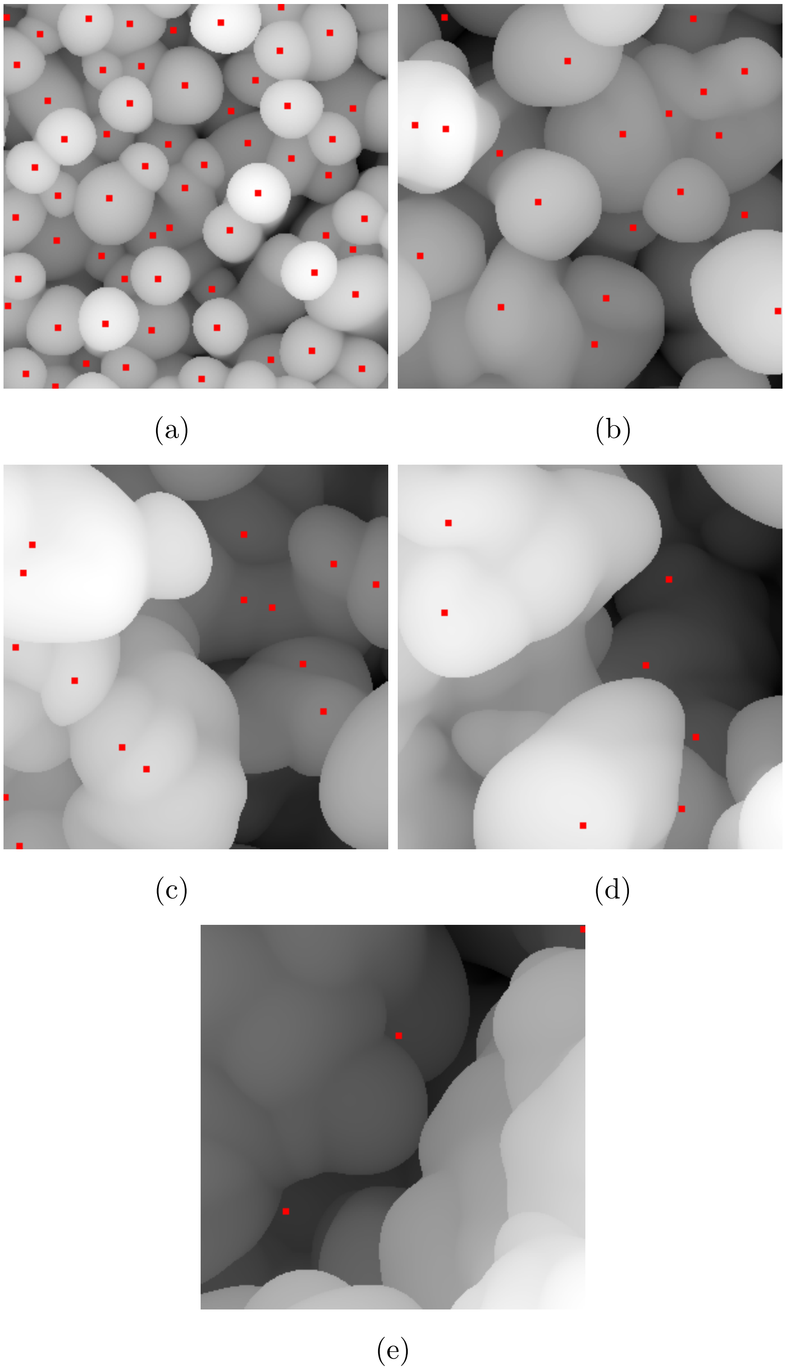


Figure 23: Bubbles identified by the magnitude of the $x-y$ velocity method, using the variance approach to identify bubble tip pixels. The images are for the central 6×6 columns of the LES data at every 100-th time step, starting with time step 100 and ending at time step 600. The results at time step 700 are similar to those at 600.



(a) (b)

(c) (d)

(e)

Figure 24: Bubbles identified by the magnitude of the $x-y$ velocity method, using the variance approach to identify bubble tip pixels. The images are for the central 300×300 pixels of the DNS data at every 50-th time step, starting with time step 50 and ending at time step 247.

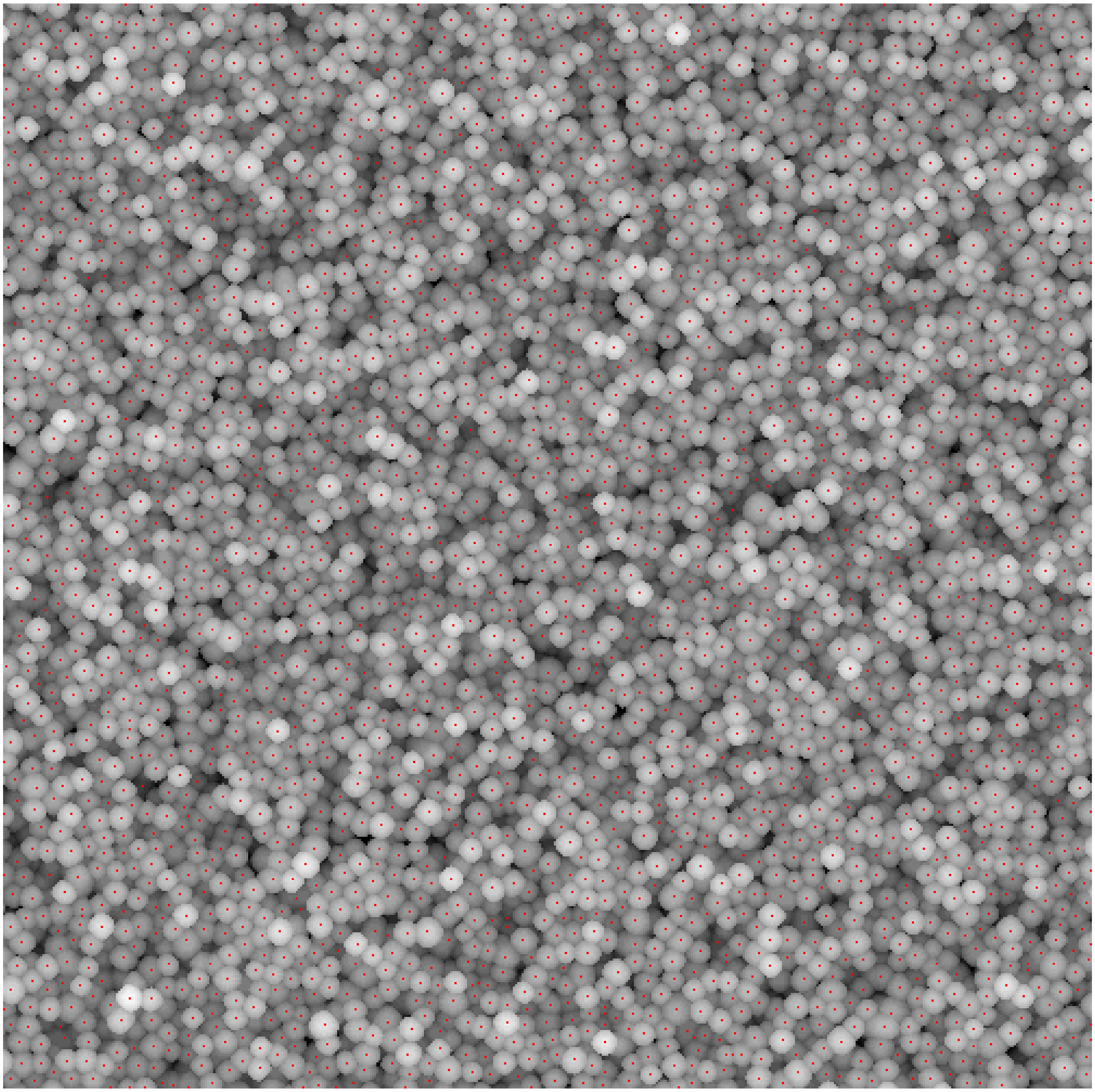


Figure 25: A 500 × 500 sub-image from time step 100 of the LES data indicating (in red) the bubble tips found by the variance-based magnitude of the $x - y$ velocity method.

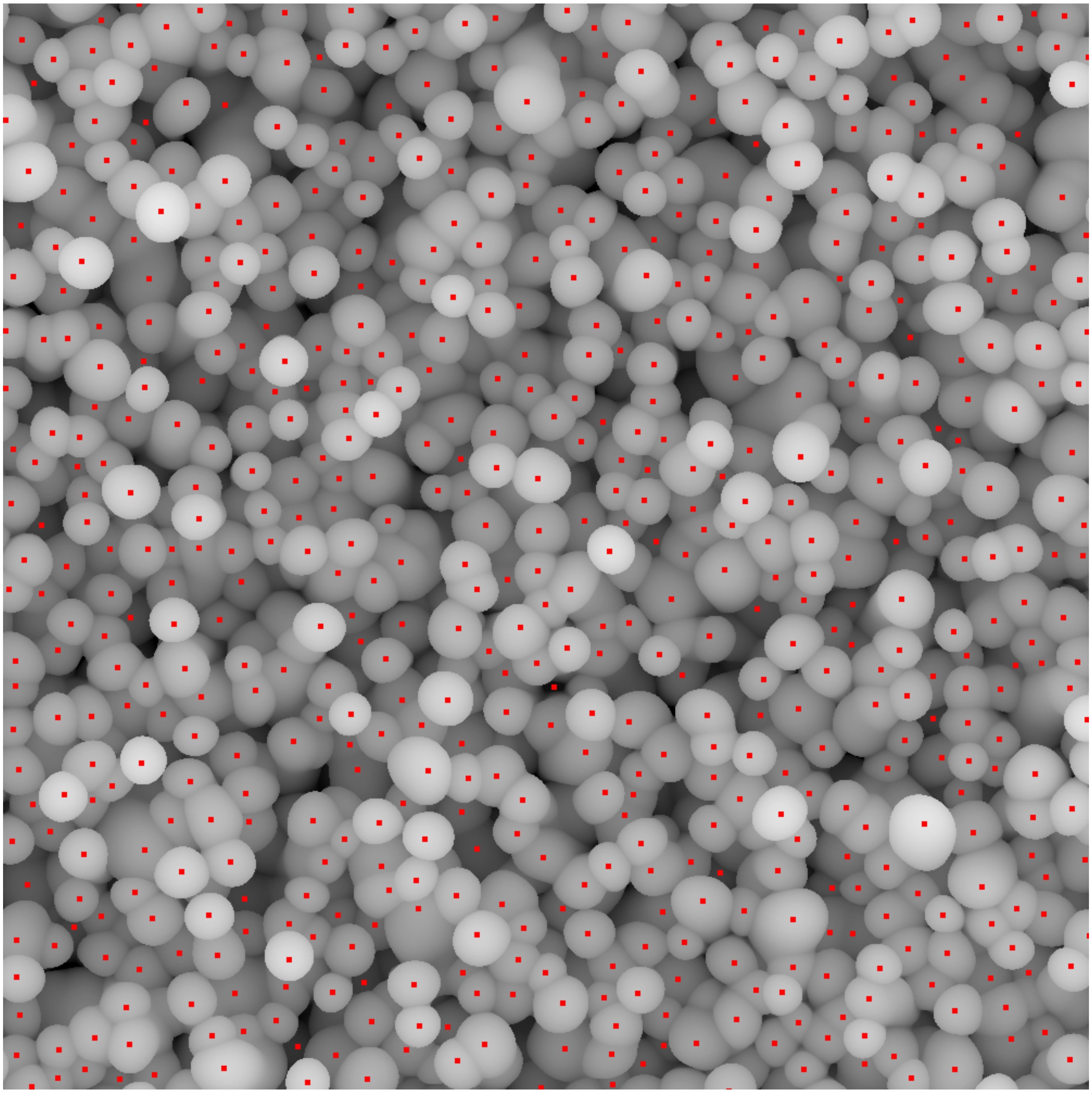


Figure 26: A 1000×1000 sub-image from time step 50 of the DNS data indicating (in red) the bubble tips found by the variance-based magnitude of the $x - y$ velocity method.

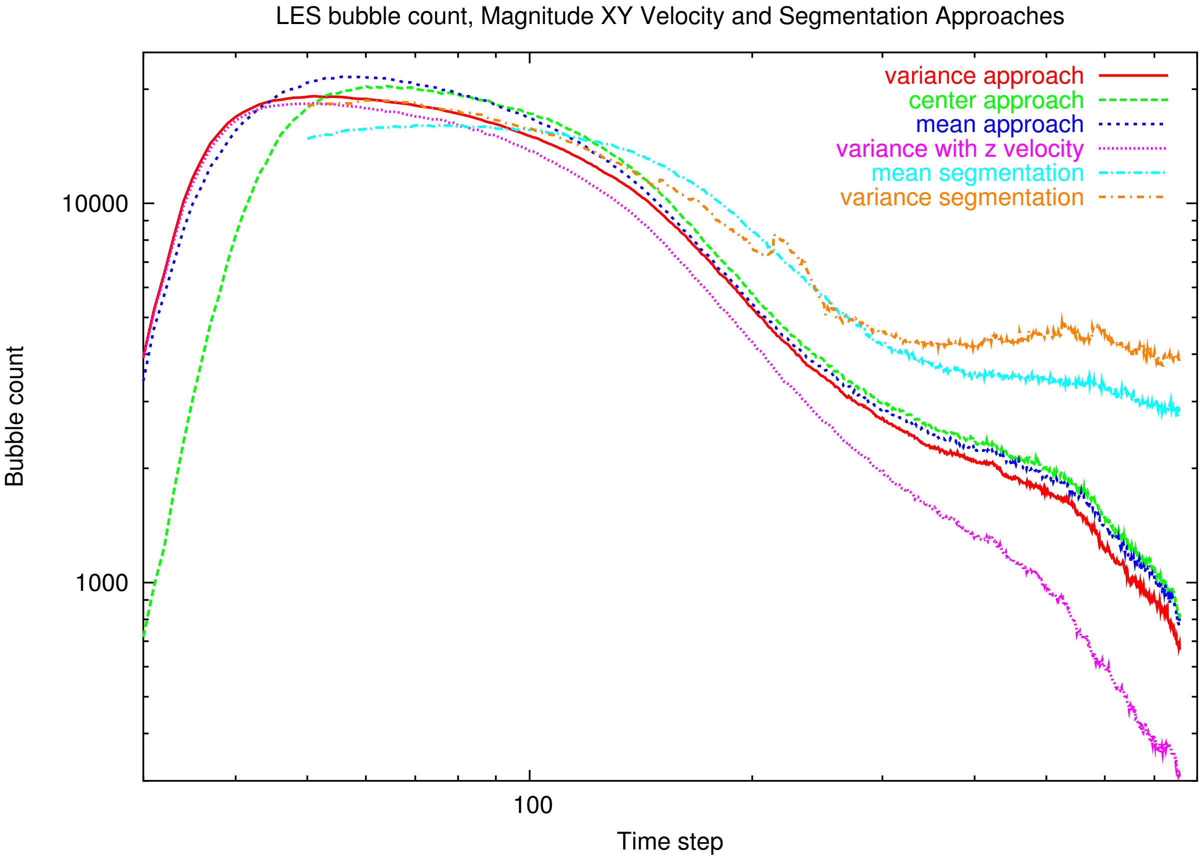


Figure 27: The bubble counts for the LES data obtained for the different methods.

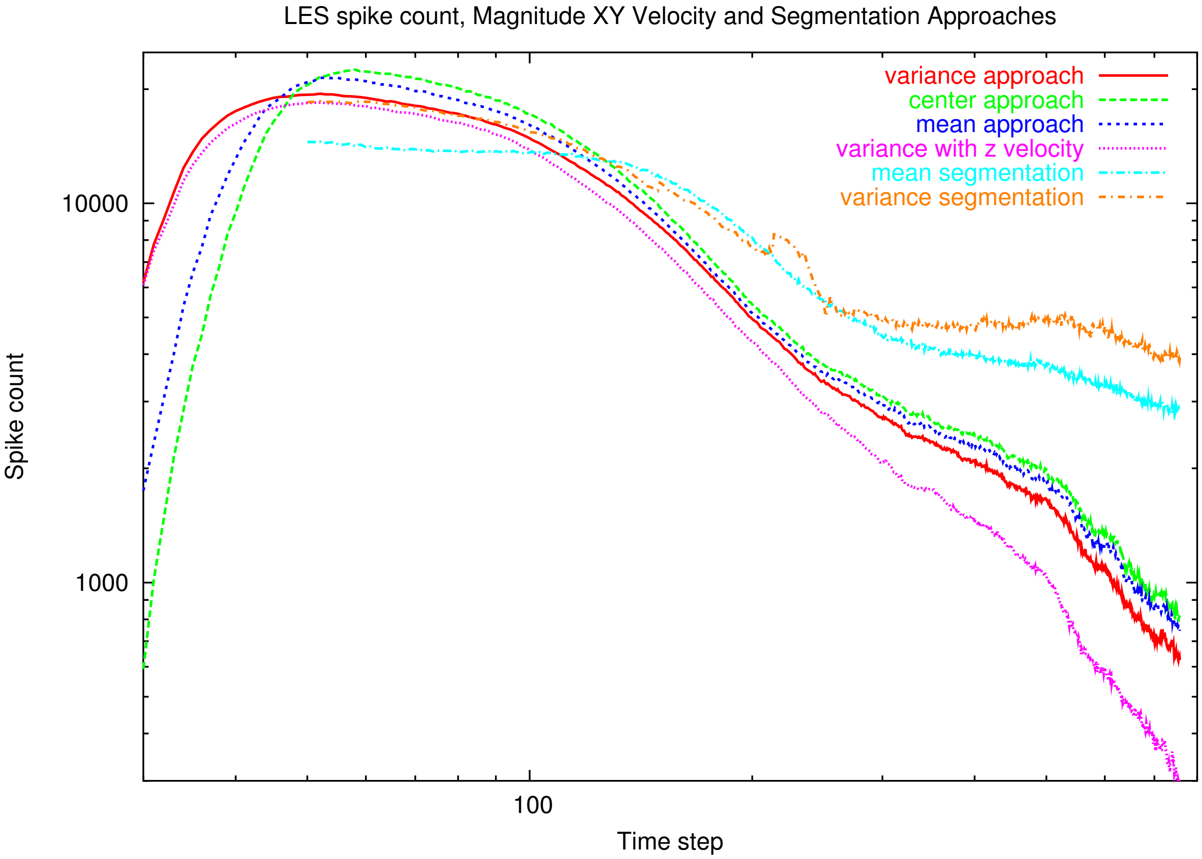


Figure 28: The spike counts for the LES data obtained for the different methods.

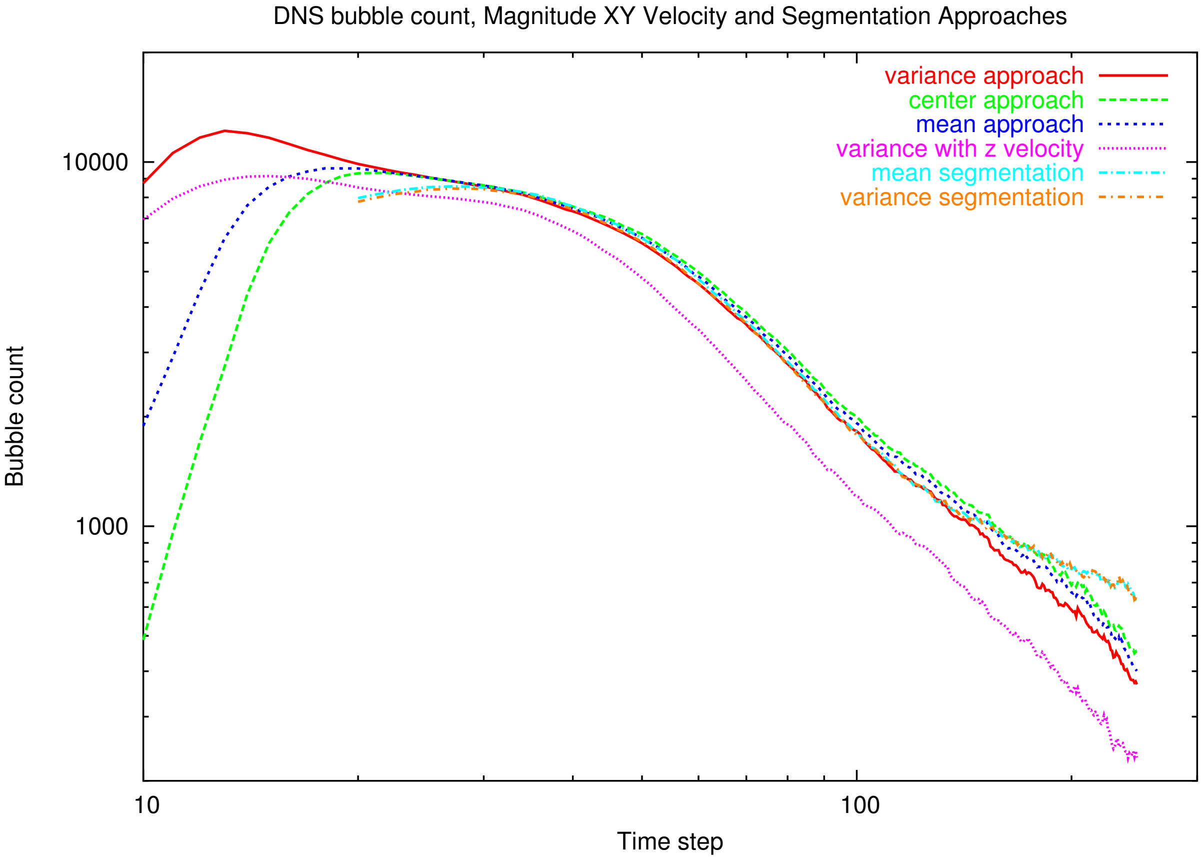


Figure 29: The bubble counts for the DNS data obtained for the different methods.

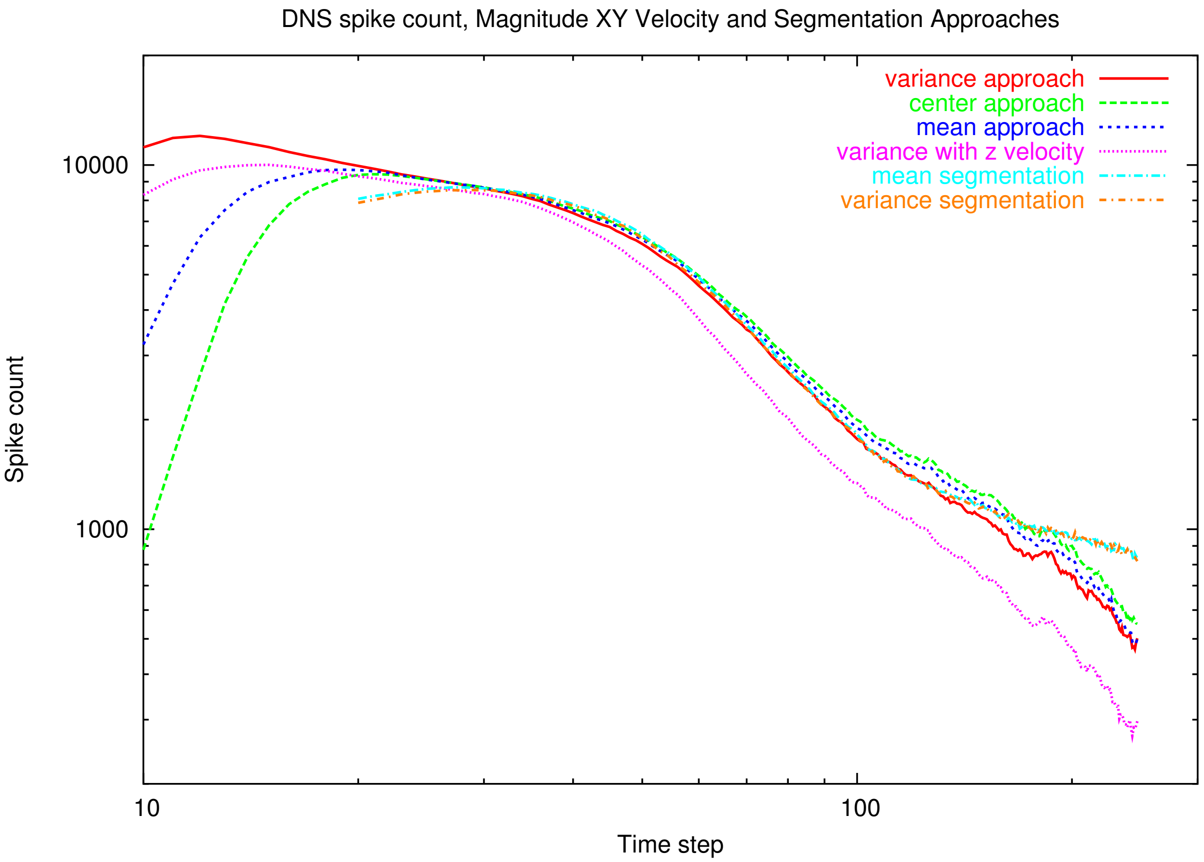


Figure 30: The spike counts for the DNS data obtained for the different methods.

to identify the pixels at the tip of the bubble/spike. As expected, this curve counts fewer bubbles and spikes, especially at the later time, where there are several bubbles/spikes which are missed as their tip does not satisfy the $z-$velocity constraint. The curve for this method however, is similar in shape to the other curves. We also observe that the techniques based on the region-growing segmentation give higher counts at late time. We expect this behavior as we are using the same threshold values during the cleanup stage for all time steps. At late time, the size threshold in the cleanup phase is such that it does not remove regions which should be removed. Recall that the region growing methods are applied to the height-depth map images. They essentially divide the images into regions, where the pixels in each region have roughly the same height. At early time, these “flat” regions are larger around the tips of the bubbles and relatively smaller elsewhere. At late time, a similar situation holds, except now all the regions are larger than at early time. So, a threshold which is set to remove the small regions at early time, will not remove the relatively small regions at late time. If the threshold is set to remove the small regions at late time, then it will also remove the bubble tip regions at early time. This indicates that for the cleanup phase of the region-growing segmentation, we may need to change the threshold values over time.

These observations hold true for the bubble/spike counts for the LES data as well. However, the region-growing segmentation methods over count both the bubbles and spikes starting from time step 100 onward. Further, the curves for the variance-based homogeneity metric are not as smooth as the ones for the mean-based homogeneity metric, though both have a shape similar to the other curves. The curve for the mean-based homogeneity metric also under counts at early time, especially in the case of the spikes. It appears that the region-growing methods are more sensitive to the lower resolution of the LES data.

- For bubbles and spikes in both the LES and DNS datasets, the three methods - variance, mean, and center - based on the magnitude of the $x - y$ velocity are very similar in performance. The behavior of the curves, even at small scale, is very similar, with the center approach usually giving slightly higher counts than the mean approach, which, in turn, is slightly higher than the variance-based method. The computational complexity of these methods is very similar and all are trivial to implement.

- For all methods, we do not include the very early time steps as the bubbles and spikes have barely formed in the height-depth map images, being just a couple of pixels off the initial fluid interface. For our analysis approach, we expect correct results starting at time step 50 for the LES data and 20 for the DNS data.

From the physics viewpoint, these curves for the bubble and spike counts indicate that there are four distinct regimes in the process of fluid mixing [5], indicated by the four regions with different slopes. We ignore the initial startup and consider these curves from time step 50 (20) onward for the LES (DNS) data. The first stage corresponds to linear growth, where the initial perturbations increase in magnitude but grow independent of each other. In the second non-linear stage, the surface of the bubbles and spikes is no longer single valued and some bubbles/spikes grow faster than the others. The third stage is one of mixing transition, where the fluids are not quite mixed. The final stage is one of strong turbulence, leading to well-mixed fluid.

9 Sensitivity Analysis for Various Thresholds

Our analysis of the LES and DNS datasets to count the bubbles and spikes first converted the 3-D data into 2-D and then processed the 2-D data to count the bubbles and spikes. Both these phases require various thresholds and it is obvious to ask how sensitive our results are to the choice of these thresholds. In our analysis, we have used a constant threshold for most of the bubble counting algorithms. These thresholds were selected via a visual examination to ensure that they correctly counted all the bubbles and spikes in the mid-range of the time steps. Therefore, we believe that the results from the bubble counting algorithms are reliable.

We next discuss the sensitivity of the bubble/spike count to the choice of the threshold used in converting the 3-D data into 2-D, while keeping constant the thresholds used for counting the bubbles and spikes. In the 3-D region-growing segmentation described in Section 6.2, we observed that there is a range of threshold values over which the 2-D images we obtain for the different variables contain no deformed bubbles or spikes. For both bubbles and spikes, as the threshold approaches the density of the respective fluids (that is, 3 for bubbles and 1 for spikes), the size of bubbles and spikes increases. When the size has increased to a certain point, any further increase in the threshold creates jagged bubble and spike boundaries as extra fluid is identified as part of either the bubble or the spike. When the threshold is farther away from the fluid density, the bubbles and spikes decrease in size and eventually start to deform to point the they may no longer be detected.

The thresholds we have selected in Section 6.6.1 and 6.6.2 avoid the problems of excessive blurring of the boundaries, the formation of jagged boundaries, as well as the deformation of bubbles and spikes. Since a change in this threshold changes the height-depth map image, and thus the images for the other variables at the bubble boundaries, we need to ensure that small variations in the threshold do not drastically affect these images as it will affect the number of bubbles and spikes counted.

Ideally, we should conduct the sensitivity analysis at each time step for both the LES and the DNS simulations. However, this requires a full sweep through the data, which can be prohibitively expensive as the LES data is 30TB and the DNS data is 80TB in size. We instead selected a few time steps and conducted extensive tests to quantify the sensitivity of our results to changes in the 3-D threshold. We evaluate the sensitivity of the threshold in a three part process as follows:

1. Step 1: For each of the time steps selected, we varied the threshold and visually inspected the height-depth map images. This allowed us to identify the acceptable threshold range which is defined as the range over which there are no bubbles with jagged boundaries, bubbles which are deformed, or bubbles which are too small to be consistently identified as bubbles by our algorithms.

2. Step 2: Once we determined the upper and lower threshold boundaries for both bubbles and spikes, we obtained a slice of the data from the 3-D region growing segmentation at these thresholds. This allowed us to confirm the validity of the thresholds from a side

profile of the bubble and spike boundaries. We show sample slices of these images for the DNS data in Figure 35.

3. Step 3: In the last step, we determined how the numbers of bubbles and spikes changed over the valid threshold range. To isolate the threshold sensitivity of the 3-D region growing, we need to use a bubble counting method which has fixed parameters over time, so that the results are not affected by the variation in the parameters for the counting method. For this reason we chose the variance-based magnitude of $x-y$ velocity method described in Section 7.2 applied with the parameters described in Section 8.

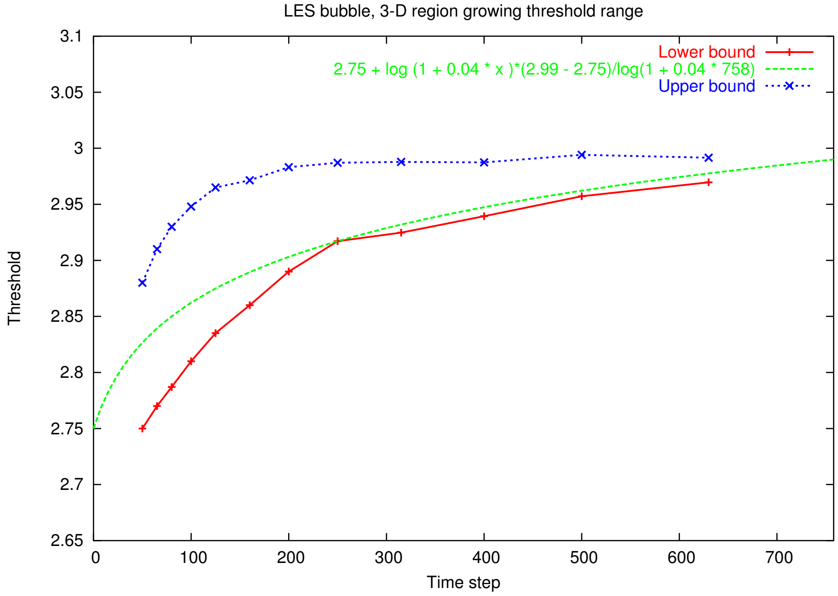
9.1 3-D Threshold Sensitivity for the LES Data

For the LES data, we performed extensive threshold tests for twelve time steps which are roughly equi-distant in the log scale. These time steps are at 50, 65, 80, 100, 125, 160, 200, 250, 315, 400, 500, and 630. Panels (a) of Figures 31 and 32 show the original thresholds for bubbles and spikes (in green), as well as the acceptable range of thresholds (in blue and red) which does not statistically change the bubble/spike count or change the number of bubbles visually. Notice that for later time steps the bounds get tighter. This is due to the fact that at the later time steps the fluids are more mixed, and a small change in the threshold could lead to a large change in the surface obtained, increasing the likelihood that not all bubbles/spikes are identified correctly. Panels (b) of Figures 31 and 32 show the bubble and spike counts, respectively, using the logarithmic thresholding described earlier, along with the counts obtained at the 12 time steps. As we can see from the results, the plots are a slight shift of each other. This is expected since the change in the threshold leads to changes in the size of the bubble and spikes. This can affect the bubble/spike count as it changes the time step when a bubble tip is covered by another or the time step where merging bubbles are identified as a single bubble.

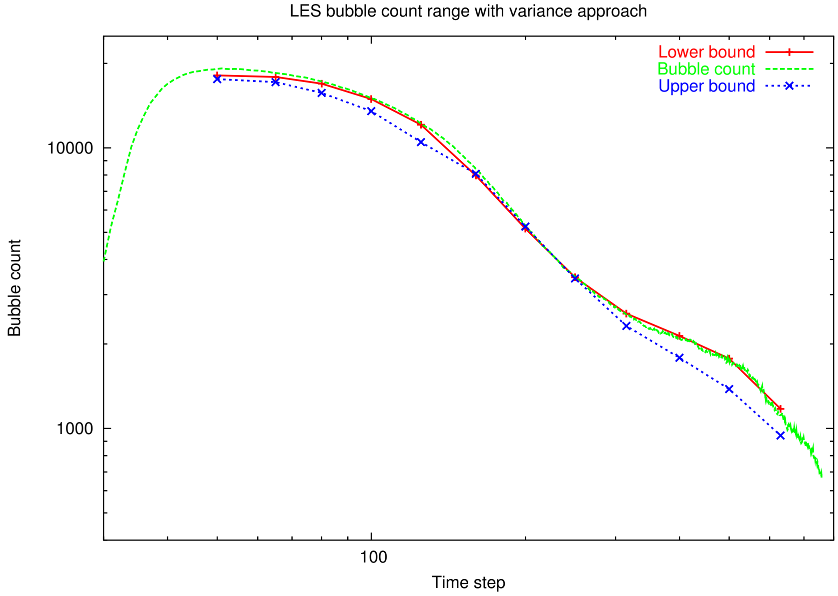
9.2 3-D Threshold Sensitivity for the DNS Data

In a manner similar to the LES data, we chose twelve time steps which are roughly equidistant in the log scale to perform an extensive threshold sensitivity test. These time steps are at 18, 22, 28, 35, 45, 56, 71, 89, 112, 141, 178, and 223. Panels (a) of Figure 33 and 34 show the logarithmic threshold (in green) we chose for the bubbles and spikes, respectively, along with the acceptable range of thresholds (in red and blue) which does not statistically change the bubble and spike counts. As with the LES results, the bounds on the threshold change are tighter at later time steps. Panels (b) of Figures 33 and 34 show the bubble and spike count, respectively, for all time steps along with the count for the twelve time steps obtained at the threshold boundaries. Note again the slight shift of the plots which is expected as explained earlier.

Figure 35 shows the thresholds superposed on a slice through the DNS data at time steps 18, 35, 56, and 89. These images confirm that the number of bubbles and spikes counted using our choice of threshold for the 3-D region growing is not sensitive to small changes in the threshold.

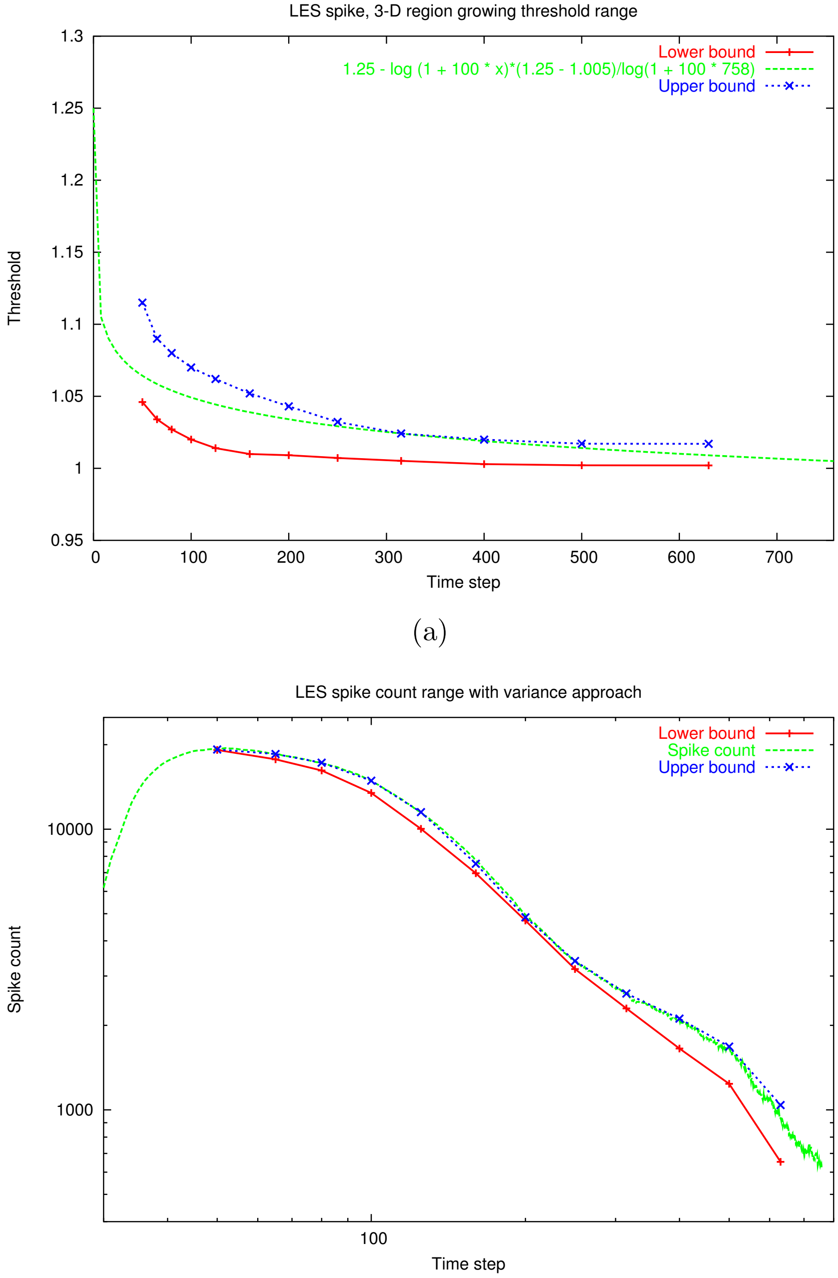


(a)



(b)

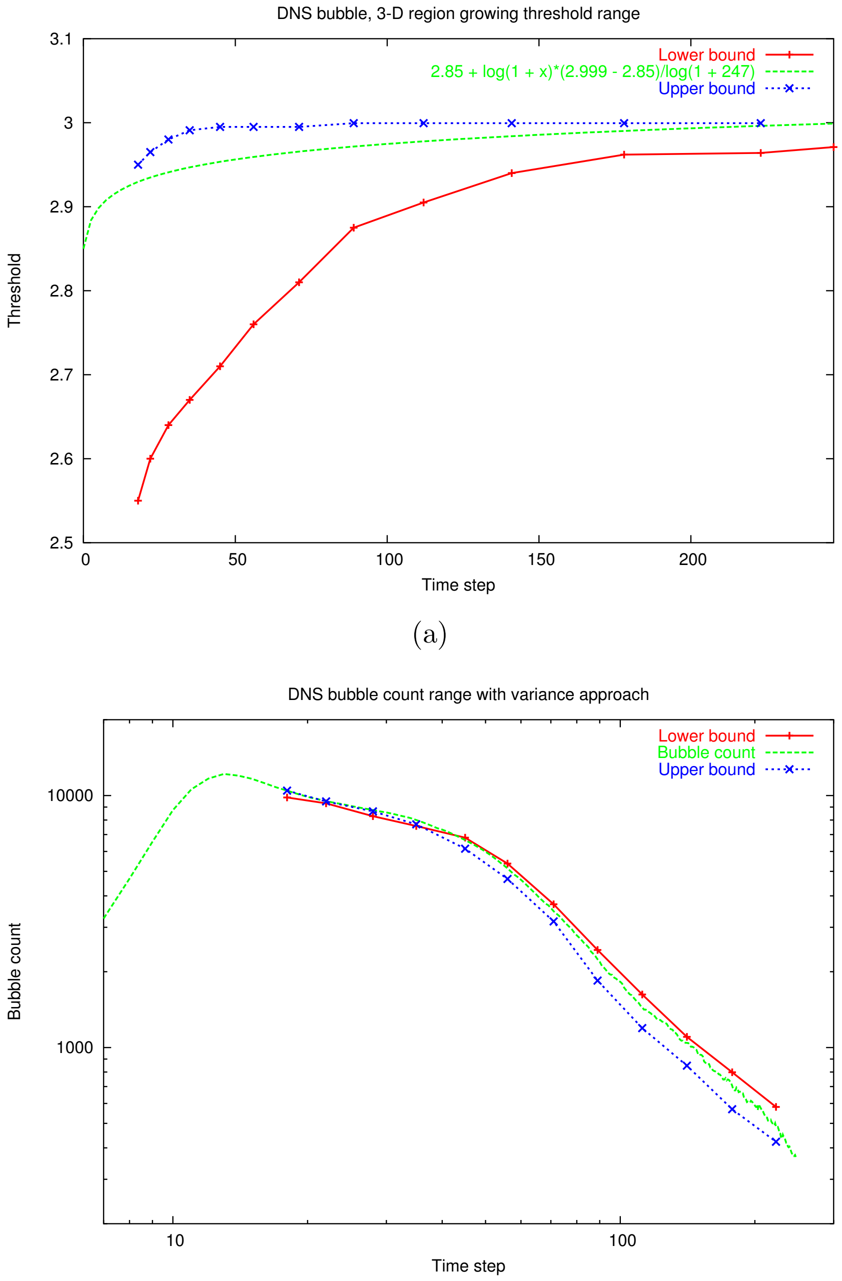
Figure 31: Threshold range of 3-D region growing for the identification of bubbles in LES data. Panel (a) shows the threshold ranges computed at the twelve equidistant log-time steps. Blue indicates the threshold upper bound, red the threshold lower bound and green the logarithmic threshold used in our work. Panel (b) shows the bubble count computed at threshold values given in (a).



(a)

(b)

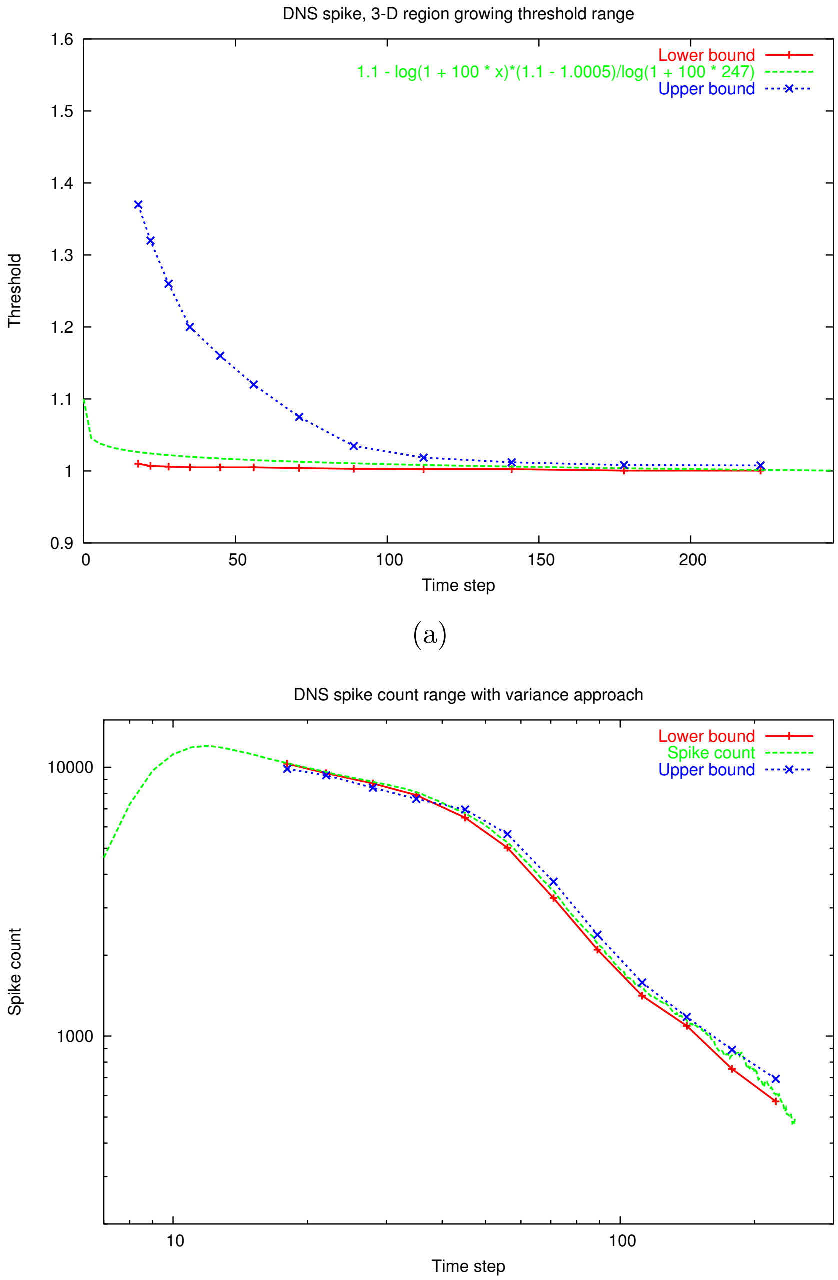
Figure 32: Threshold range of 3-D region growing for the identification of spikes in LES data. Panel (a) shows the threshold ranges computed at the twelve equidistant log-time steps. Blue indicates the threshold upper bound, red the threshold lower bound and green the logarithmic threshold used in our work. Panel (b) shows the spike count computed at threshold values given in (a).



(a)

(b)

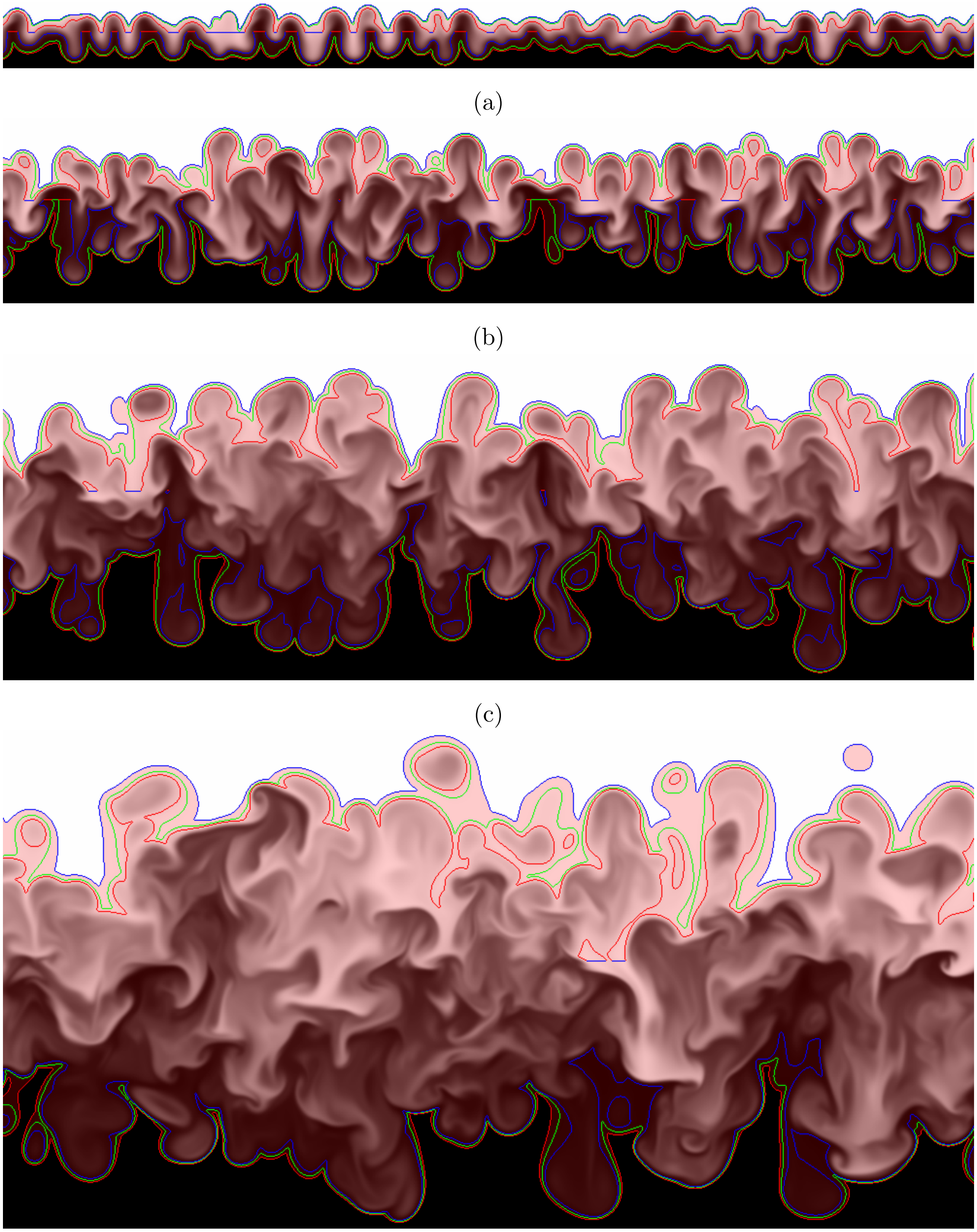
Figure 33: Threshold range of 3-D region growing for the identification of bubbles in DNS data. Panel (a) shows the threshold ranges computed at the twelve equidistant log-time steps. Blue indicates the threshold upper bound, red the threshold lower bound and green the logarithmic threshold we use. Panel (b) shows the bubble count computed at threshold values given in (a).



(a)

(b)

Figure 34: Threshold range of 3-D region growing for the identification of spikes in DNS data. Panel (a) shows the threshold ranges computed at the twelve equidistant log-time steps. Blue indicates the threshold upper bound, red the threshold lower bound and green the logarithmic threshold we use. Panel (b) shows the spike count computed at threshold values given in (a).



(a)

(b)

(c)

(d)

Figure 35: Images of a slice of the DNS data at $x = 1$ at time steps (a) 18, (b) 35, (c) 56, and (d) 89. Only the central 1000 of the 3072 pixels are displayed. The blue, green and red lineouts correspond to the spike and bubble boundaries found using the 3-D region growing method with the corresponding color of threshold in Figures 33 and 34.

10 Conclusions and Future Work

In this report, we described the use of simple image processing techniques to count bubble and spike structures in two simulations of the Rayleigh-Taylor instability. We discussed how we apply our techniques to massive data sets where each time step in the simulation is stored in multiple files. We also showed how we could make the analysis tractable by first converting the 3-D data to 2-D to reduce its size, and then analyzing the 2-D data. We described several ways of defining bubble/spike tips and compared their performance, both visually, and as plots of bubble/spike counts over the entire simulation. We showed that our results are relatively insensitive to the choice of the parameters used in the analysis algorithms. As expected, our analysis of the bubble/spike count curves also indicated that there are four distinct regimes in the process of the mixing of the two fluids.

There are several ways in which we plan to extend this work. In this report, we have assumed that a single algorithm, with a fixed set of parameters, will correctly count the bubbles and spikes over all time steps. Our results indicate that this is not entirely correct at the very early and at late time, where there is an issue of how best to define a bubble. We will address this in our follow-on work. First, we will refine the definition of bubbles and spikes at later time so that we can adequately capture the extent of the structures and the results are a more accurate reflection of the data. We will also extract other statistics such as the distribution of the size of the structures and the distances between neighboring structures. Finally, we will study the dynamics of the structures as they evolve over time.

11 Acknowledgment

We would like to thank the team at Livermore Computing for their support, especially their help with the data management issues associated with the very large datasets. The data used in this work was generated by Andrew Cook and William Cabot, who graciously made it available for our analysis. William Cabot also provided comments on an early version of this report. We also acknowledge the Sapphire project team who contributed to parts of the software used in this analysis and Rebecca Springmeyer and Jeffery Long for their encouragement of this work.

This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

References

[1] Banerjee, A., Hirsh, H., and Ellman, T. Inductive learning of feature tracking rules for scientific visualization. In *Proceedings of the IJCAI-95 Workshop on Machine Learning in Engineering* (1995).

[2] Cabot, W. H., and Cook, A. W. Reynolds number effects on Rayleigh-Taylor instability with possible implications for type Ia supernovae. *Nature Physics 2* (2006), 562–568.

[3] Canny, J. A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence 8*, 6 (1986), 679–698.

[4] Cheng, B., Glimm, J., and Sharp, D. H. A three-dimensional renormalization group bubble merger model for Rayleigh-Taylor mixing. *Chaos 12*, 2 (2002), 267–274.

[5] Cook, A. W., Cabot, W. H., and Miller, P. L. The mixing transition in Rayleigh-Taylor instability. *Journal of Fluid Mechanics 511* (2004), 333–362.

[6] Dimonte, G., et al. A comparative study of the turbulent Rayleigh-Taylor instability using high-resolution three-dimensional numerical simulations: The Alpha-Group collaboration. *Physics of Fluids 16*, 5 (2004), 1668–1693.

[7] Dimonte, G., and Schneider, M. Density ratio dependence of Rayleigh-Taylor mixing for sustained and impulsive acceleration histories. *Physics of Fluids 12*, 2 (2000), 304–321.

[8] Farge, M. Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics 24* (1992), 395–457.

[9] Farge, M., Kevlahan, N., Perrier, V., and Goirand, E. Wavelets and turbulence. *Proceedings of the IEEE 84*, 4 (1996), 639–669.

[10] Farge, M., and Schneider, K. Analyzing and compressing turbulent fields with wavelets. Tech. rep., Institute Pierre Simon de Laplace, June 2002. Available at `http://monteverdi.ens.fr/`.

[11] Ferre-Gine, J., Rallo, R., Arenas, A., and Giralt, F. Identification of coherent structures in turbulent shear flows with a fuzzy ARTMAP neural network. *International Journal of Neural Systems 7*, 5 (1996), 559–568.

[12] Hangan, H., Kopp, G. A., Vernet, A., and Martinuzzi, R. A wavelet pattern recognition technique for identifying flow structures in cylinder generated wakes. *Journal of Wind Engineering and Industrial Aerodynamics 89* (2001), 1001–1015.

[13] Oron, D., Along, U., and Shvarts, D. Scaling laws of the Rayleigh-Taylor ablation from mixing zone evolution in inertial confinement fusion. *Physics of Plasmas 5*, 5 (1998), 1467–1476.

[14] Sharp, D. H. An overview of Rayleigh-Taylor instability. *Physica 12D* (1984), 3–18.

[15] Shvarts, D., Alon, U., Ofer, D., McCrory, R., and Verdon, C. P. Nonlinear evolution of multi-mode Rayleigh-Taylor instability in two and three dimensions. *Physics of Plasmas 2*, 6 (1995), 2465–2472.

[16] Siegel, A., and Weiss, J. B. A wavelet-packet census algorithm for calculating vortex statistics. *Physics of Fluid 9*, 7 (1997), 1988–1999.

[17] Sonka, M., Hlavac, V., and Boyle, R. *Image Processing, Analysis, and Machine Vision.* PWS Publishing, 1999.

[18] Zabusky, N. Vortex paradigm for accelerated inhomogeneous flows: Visiometrics for the Rayleigh-Taylor and Richtmyer-Meshkov environments. *Annual Review of Fluid Mechanics 31* (1999), 495–536.