Automatic Segmentation of CT Scans of Checked Baggage

David F. Wiley, Deboshmita Ghosh, and Christian Woodhouse

Abstract—**We adapted our novel segmentation method, called Stratovan Tumbler, to automatically segment objects from CT scans of checked baggage at airports. Our segmentation method handles problems with CT scans such as streaking and noise and is able to make difficult splitting and merging decisions for touching objects. Our methodology produces an object hierarchy describing the relationship of baggage objects to one another that can be used during the explosive detection stage in airport security for the purpose of identifying threats. The main benefits of our method are (i) tolerance of CT reconstruction artifacts, (ii) works well on objects of any shape, and (iii) can handle both homogeneous and heterogeneous materials.**

Index Terms—**Segmentation, Automatic object segmentation, Computed tomography, Baggage screening, Airport security, Automated threat recognition, Explosive detection**

I. Introduction

EXPLOSIVE detection in airport security involves automatically analyzing x-ray computed tomography (CT) scans of checked baggage [6]. The goal of this detection process is to identify possible explosive material signatures within baggage contents prior to their being transported on a plane. The detection process involves numerous steps, with one possible step being that of image segmentation to delineate individual objects, such as a bottle of liquid, so that object characteristics can be extracted and fed into a threat recognition system to determine whether or not the object matches an explosive signature [5]. The challenge in this environment is that x-ray CT scans inherently suffer from image quality problems such as low-resolution, noise, poor contrast, and streaking artifacts to name a few. These problems adversely affect the segmentation step by, for example, incorrectly merging touching objects or by splitting an object into too many pieces to the extent that extracted characteristics are compromised to the subsequent explosive detection step.

We adapted a medical segmentation technology called Stratovan Tumbler [7] to the problem of delineating objects from CT scans of checked baggage. Tumbler is a flexible region-growing segmentation method which delineates objects irrespective of shape, topology, and orientation. We made no assumption in object shape, type, density, or composition in an effort to develop a robust system that could automatically delineate most objects. The result of this effort is Stratovan Decorum, a software tool for automatically delineating and analyzing objects from CT scans of baggage. Our work in this area was initiated by the Department of Homeland Security and focuses on improving segmentation of all objects above a minimum intensity threshold, 500 modified Hounsfield units (MHU), where water is calibrated at 1000 MHU [1].

II. Tumbler Segmentation

A. Overview

Tumbler uses a kernel that virtually moves in three dimensions in a flood-fill manner. The kernel starts inside the object and iteratively moves outward until it meets the object's boundary. Tumbler can also simulate the outside-inward process with proper setup.

Our algorithm begins by defining the kernel shape and size, and positioning it inside the object being segmented. We move the kernel outward in a flood-fill fashion while the newly traversed voxel intensities continue to satisfy predetermined criteria (e.g., mean value is above some threshold). The size and shape of the kernel prevents it from escaping through holes in ill-defined boundaries.

B. Process Details

We implement the Tumbler process by:

1. **Defining a 3D kernel**: in most cases, a sphere suffices. The kernel is always smaller than the object to be segmented and bigger than any expected holes in the object's boundary.
2. **Determine movement criteria**. Each time the kernel moves, we pre-compute the minimum, maximum, mean, and standard deviation **{min, max, mean, std}** of the "to be" traversed voxels. Based on these parameters, one must determine the criteria required to decide whether or not to accept or reject the new position. This can be as simple as comparing the mean value to a constant threshold or more complicated as needed.
3. **Choose a start location**. This can either be done manually or by some automatic process. Once established, the

Manuscript received June 15, 2012. This work was funded by the Northeastern University ALERT program under the CT Segmentation Initiative.

D. Wiley is the founder and CEO of Stratovan Corporation, Sacramento, CA 95826, USA (phone: 916-813-7233; e-mail: wiley@stratovan.com).

D. Ghosh is with Stratovan Corporation, Sacramento, CA 95826, USA (email: dghosh@stratovan.com).

C. Woodhouse is with Stratovan Corporation, Sacramento, CA 95826, USA (email: woodhouse@stratovan.com).

neighboring voxel coordinates and direction of movement from the current position *{i, j, k, dir}* are placed in a queue that drives the flood-fill process.

4. **Flood-fill process**:
   a. Remove a position from the queue and compute the **{min, max, mean, std}** values of the voxel intensities.
   b. Compare the **{min, max, mean, std}** to the criteria established in Step 2 and decide whether these new voxels are acceptable or not.
   c. If not acceptable, do nothing more. If acceptable, move the kernel to this new position, mark all voxels covered by the kernel as the "object," and add all of its non-traversed neighbors to the flood-fill queue.
   d. Repeat Step 4a.

5. **Completion**. The process stops when the flood-fill queue is empty. The voxels traversed by the kernel represent the object.

This process is illustrated in Fig. 1.

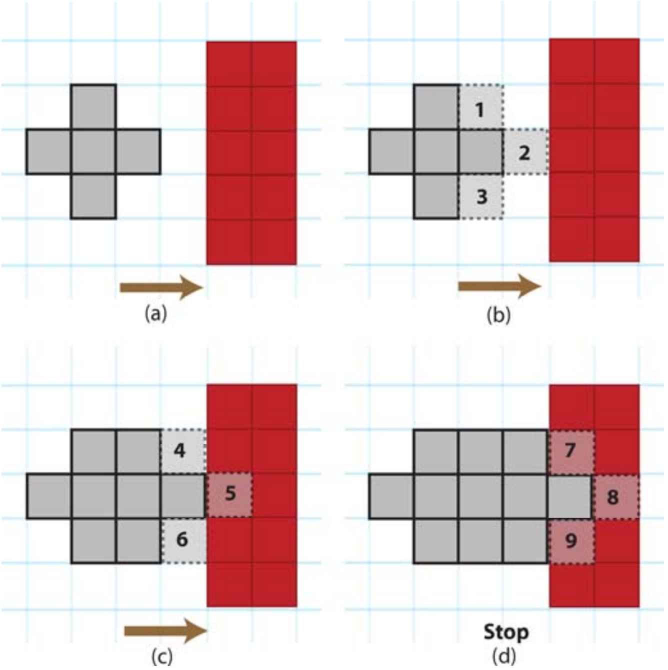


Fig. 1. Illustration of the Tumbler process. In each image, red pixels indicate the object's boundary while the white pixels indicate the object. Image (a) shows the initial condition having a "+" shaped kernel, as indicated by the gray pixels. Image (b) shows that to move the kernel one position to the right, pixels 1, 2, and 3 must satisfy some predetermined criteria; such as their average value must be below some threshold. Assuming that this criteria is met, image (c) indicates the completed move and tries another by evaluating the intensity values of pixels 4, 5, and 6. Assuming that the average of these intensity values continues to satisfy the guiding criteria, image (d) indicates the completed move and now evaluates the average intensity of pixels 7, 8, and 9. In case (d) we find that the average value is now above our threshold, preventing further movement to the right. (The algorithm would then continue in the remaining directions.)

C. *Algorithm Variables*

Parameters that control the Tumbler process are:

- **Kernel size and shape**. A bigger kernel is more tolerant to noise but can consume fine detail internal to an object and at its boundary. A smaller kernel results in better detail but is susceptible to noise and, if objects are touching, the flood-fill process can spill into neighboring objects. The rule of thumb is that the kernel must be smaller than the object of interest yet bigger than any holes in its boundary.
- **Start position**. The tumbler kernel must start inside an object of interest. Choice of start location within the object can affect segmentation results in some cases and must be evaluated on a case-by-case basis during segmentation training in order to determine optimal starting position requirements.
- **Guiding criteria**. Each object of interest must be evaluated to determine acceptable parameters for **{min, max, mean, std}** voxel intensities. This information constrains the flood-fill movement of the kernel at each iteration.
- **Voxel spacing**. Tumbler performs better with near uniform voxel spacing. Additionally, Tumbler performs substantially better with a kernel larger than a single voxel. The spacing between each voxel imposes a lower bound on the kernel size and ultimately on the smallest extracted objects. Resampling data to a finer resolution can, in some cases, better identify thin objects.

D. *Adapting Tumbler to Aviation Security*

Our focus during the project period with the Northeastern University ALERT CT Segmentation Initiative was to develop a fully automated system for segmenting baggage scans. This involved automating various steps and refining techniques used at various stages to improve segmentation quality. These improvements are described in the following sections.

1) *Algorithm Automation*

We devised methods for automatic selection of Tumbler parameters in order to automatically analyze baggage scans:

Kernel Shape and Size: We were able to devise a robust method for automatically choosing a kernel size for any point within a CT scan. This is important since we must choose a kernel size that is smaller than the object being delineated yet larger than any expected holes in the object boundary. We were able to determine kernel size (1-, 2-, 3-, or 4-mm radius sphere) based on an analysis of image gradient near each voxel, which implies the amount of "clutter" at any given point. More clutter demanded a smaller kernel while less clutter permitted a larger kernel since the surrounding area represented a relatively homogeneous region.

Start Position: We were able to devise an ordering method for each voxel to choose seed points for starting each object delineation process. We found that considering the following circumstances enabled the automatic determination of seed point priority:

- Run large kernel sizes first and smaller sizes last.
- Run high-intensity voxels first and low-intensity voxels last.
- Start in the middle of objects rather than on the edges.

- Increase the priority of voxels within "thin" objects, since thins objects tended to be sorted last, but we still needed to delineate them.

Guiding Criteria: We setup a training process for determining guiding criteria used when moving the kernel from one voxel to the next. Initially, this consisted of choosing minimum and maximum thresholds that bounded the mean intensity computed from the kernel when placed at each seed point (i.e., choosing 0.98*mean and 1.02*mean for the minimum and maximum thresholds, respectively). We were able to mature this process by manually improving errant segmentations (by either increasing or decreasing the thresholds) and adding each improvement to a central training file which eventually replaced our initial, crude estimate. We fit a polynomial function to the training points while considering mean intensity, standard deviation, minimum intensity, maximum intensity, and gradient information within the kernel as placed at the seed location. The trained polynomial was then used to automatically determine guiding criteria at any voxel location within the CT scan. Furthermore, we tracked error metrics associated with the training function to maintain integrity of the system. In general, we found that the training process converged quickly with minimal to no un-intended side effects as training points were added. In total, we used just over seventy training points from around twenty training bags.

2) *Splitting*

We found that our method works rather well at splitting objects into distinct and relatively homogeneous parts. We also found that our training process converged quickly when correcting splitting errors and was able to separate almost any touching objects if there existed a gradient boundary between them. Also, in many cases, *our system is able to separate touching objects when there does not exist a gradient boundary*. This is possible with our multi-voxel kernel since the larger kernel does not slip through the touching region. By manipulating the kernel size and the guiding criteria we are able to separate touching objects, for example, a lotion and gel pad, as shown in Fig. 2.

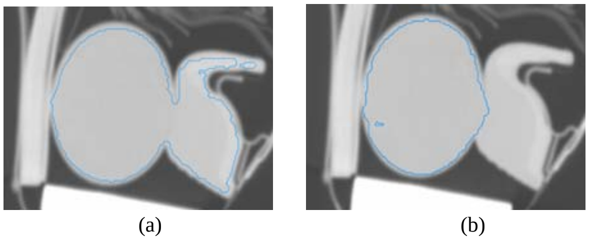


Fig. 2. Segmentation result of two touching objects (bottle of lotion and gel pad) with kernel radius 2mm (a) and 4mm (b). By using a larger kernel we are able to identify the slight gradient boundary between the two objects, as indicted by the blue outline.

3) *Merging*

Our process tends to split objects into relatively homogeneous parts. For example, a cell phone is split into its internal electronics, plastic case, leather case, and belt buckle. This introduces the problem of *object philosophy*. The object philosophy problem is simply: how does one represent something like a cell phone in their segmentation results: *Is it a single object or is it multiple component parts?*

By determining a tree hierarchy that defines how each part relates to the others, the object philosophy problem is averted since a cell phone exists both as component parts and a single composite object depending upon how one traverses the parts tree.

We computed a hierarchical representation of segmented parts by evaluating overlap between each delineated part and *merging* two parts in our tree that meet certain criteria. (We initially based this decision on whether two parts overlapped by 10% or more voxels which worked well for most cases.) Our merging process takes advantage of a side-effect of our Tumbler segmentation which results in overlapping object segmentations, as indicated by the arrows in Fig. 3.

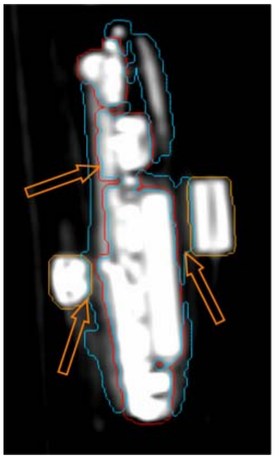


Fig. 3. Segmentation of a cell phone into overlapping component parts: internal electronics, plastic case, belt buckle, etc. Arrows indicate overlapping voxels between parts implying connectedness.

Fig. 4 shows the merged components of a candle (wax inside a metal container with lid) and a camera tripod (constituting over forty individually segmented parts).

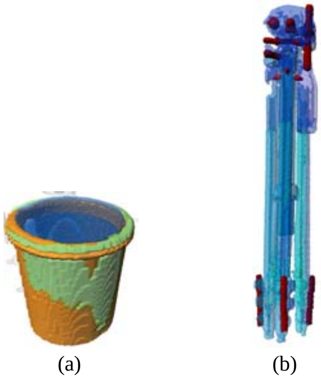


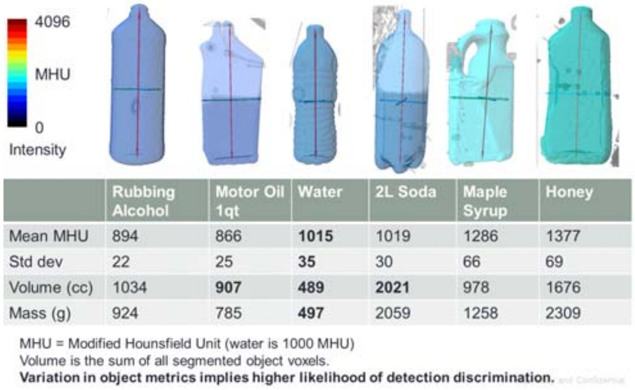
Fig. 4 Results of our merging process on a candle (a) and a tripod (b).

III. Results

We tested our software on over thirty baggage CT scans provided to us by the Northeastern University ALERT CT

Segmentation Initiative [1]. Each scan was captured on a medical grade CT scanner and had a resolution of 0.98 x 0.98 x 1.29 mm with water calibrated at a value of 1000 MHU.

Our Tumbler segmentation method is effective in delineating many different kinds of objects from CT scans of baggage. Bulk or homogeneous objects such as a bottle of water, honey, rubbing alcohol, hydrogen peroxide, nylon phantom, and rolls of tape are generally composed of a single material. We are able to segment nearly all objects in this category accurately, as shown in Fig. **5** and Fig. **6**.



	Rubbing Alcohol	Motor Oil 1qt	Water	2L Soda	Maple Syrup	Honey
Mean MHU	894	866	**1015**	1019	1286	1377
Std dev	22	25	**35**	30	66	69
Volume (cc)	1034	**907**	**489**	**2021**	978	1676
Mass (g)	924	785	**497**	2059	1258	2309

MHU = Modified Hounsfield Unit (water is 1000 MHU)
Volume is the sum of all segmented object voxels.
Variation in object metrics implies higher likelihood of detection discrimination.

Fig. 5. Segmentation result of liquids with varying intensities. Examples include bottles of rubbing alcohol, motor oil, water, honey, soda, maple syrup, and honey. The mean intensities, volume, and mass are computed from our segmentation results.

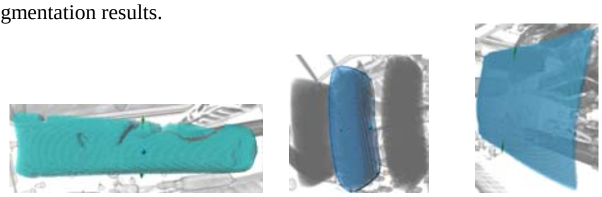

(a) (b) (c)

Fig. 6. Segmentation results of (a) toothpaste, (b) soap and (c) rubber sheet. We are able to delineate all objects that are generally composed of a single material with reasonably well defined boundaries (even if these objects are touching other objects).

We are able to easily separate touching objects since we use a large kernel size allowing us to effectively detect ill-defined object boundaries (i.e., the touching region even when there is no gradient boundary between objects).

In the case of heterogeneous objects, including cell phones, laptops, hard drives, etc., we delineate the components parts of the object and merge those together into a parts tree describing the connectedness of the parts (see Fig. **4** and Fig. **7**).

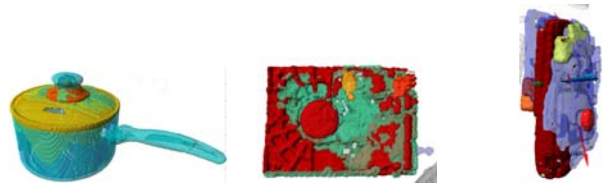

(a) (b) (c)

Fig. 7. Segmentation results of (a) metal pot with lid, (b) harddrive and (c) cell phone. Objects that are composed of heterogeneous parts (i.e., materials of different density) are delineated as separate parts and later combined in the merging step.

Thin objects with low intensity are poorly represented by CT due to lower contrast at the lower intensity. Sometimes these objects touch high-intensity objects like the neoprene sheet sitting on top of a Costco package of 48 AA batteries in Fig. **8**, which destroys the gradient boundary information making it challenging for any segmentation method to delineate accurately. Such CT artifacts cause segmentations to result in multiple parts, which we recombine later in the merging step.

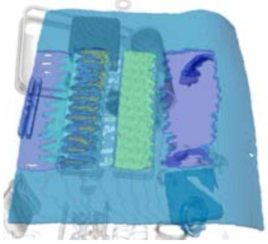


Fig. 8. Segmentation result of neoprene sitting on top of a package of 48 AA batteries. Here we are unable to delineate the sheet into a single segmentation since streak artifacts from the high-intensity batteries destroy the gradient boundary and erroneously changing the neoprene intensity in various regions.

IV. Conclusions

Tumbler segmentation tolerates common problems inherent in image segmentation and those caused by CT reconstruction artifacts such as noise, streaking, shading, and poor representation of object boundaries. It employs a multi-voxel kernel which considers several voxel intensities when deciding whether to connect neighboring voxels in a region-growing step. This allows the Tumbler method to tolerate noise, streaks, shading, and most importantly to detect ill-defined object boundaries in the case of touching objects.

We are currently improving performance of our Tumbler algorithm (currently runs about 40 min) by adapting it to the GPGPU and also improving our parts tree merging process. We are also investigating the impact our improved segmentation technique has on explosive detection quality.

References

[1] C.R. Crawford, H.E. Martz, and H. Pien, "Segmentation of Objects from Volumetric CT Data," Northeastern University, Boston. This report can be found at: https://myfiles.neu.edu/groups/ALERT/strategic_studies/SegmentationInitiativeFinalReport.pdf

[2] Z. Lin, J. Jin, and H. Talbot, "Unseeded region growing for 3D image segmentation," Selected papers from Pan-Sydney Workshop on Visual Information Processing, Sydney, Australia, 2001.

[3] K.R. Peschmann and J. Harmann, "Automatic Concealed Object Detection System Having a Pre-Scan Stage," US Patent 5,182,764, January 26, 1993.

[4] T. Shen, H. Li, Z. Qian, and X. Huang, "Active Volume Models for 3D Medical Image Segmentation," Computer Vision and Pattern Recognition, 707-714, 2009.

[5] S. Simanovsky, I.M. Bechwati, M. Hiraoglu and C.R. Crawford, "Apparatus and method for detecting sheet objects in computed tomography data," US Patent 6,026,143, February 15, 2000.

[6] S. Singh and M. Singh, "Explosives detection systems (EDS) for aviation security," Signal Processing 83 (2003), 31-55.

[7] D. Wiley, "Analysis of Anatomic Regions Delineated from Image Data," US Patent 8,194,964 (US App. No. 12/430,545), filed April 27, 2009.