34th Annual International Conference of the IEEE EMBS
San Diego, California USA, 28 August - 1 September, 2012

Detecting Distance between Injected Microspheres and Target Tumor via 3D Reconstruction of Tissue Sections*

James P. Carson, *Member, IEEE,* Andrew P. Kuprat, Sean M. Colby, Cassi A. Davis, Christopher A. Basciano, Kevin Greene, John T. Feo, Andrew Kennedy

Abstract— **One treatment increasing in use for solid tumors in the liver is radioembolization via the delivery of ^{90}Y microspheres to the vascular bed within or near the location of the tumor. It is desirable as part of the treatment for the microspheres to embed preferentially in or near the tumor. This work details an approach for analyzing the deposition of microspheres with respect to the location of the tumor. The approach used is based upon thin-slice serial sectioning of the tissue sample, followed by high resolution imaging, microsphere detection, and 3-D reconstruction of the tumor surface. Distance from the microspheres to the tumor was calculated using a fast deterministic point inclusion method.**

I. Introduction

Solid tumors in the liver, both primary and metastatic, are common and challenging for an oncologist. Despite various treatment options, mortality rates remain high, with these tumors accounting as the main cause of death for approximately 80,000 patients in the United States each year [1]. Hepatocellular carcinoma (HCC) is increasingly recognized and incidence is growing. ^{90}Y microsphere therapy delivers high numbers of microspheres with resulting high total doses of radiation. ^{90}Y is a beta emitting isotope that delivers high doses of radiation up to 3mm from the sphere, with a half life of 64 hours [2,3].

Understanding the distribution of microspheres relative to the tumor site is important to determining the amount of radiation delivered to the tumors and nearby tissues. Here we demonstrate a new approach developed for measuring distance between microspheres and a tumor. The steps involved are: tissue acquisition; tissue embedding and serial sectioning; digitizing tissue at high resolution; labeling the images to mark microsphere locations and tumor regions; registration of the 2D image series to remove distortions caused by the sectioning process and to create a coherent 3D tumor volume; and calculation of distances between microspheres and tumors using the Khamayseh-Kuprat method [4].

*Research supported by Sirtex Medical Inc., by the U.S. Department of Energy under LDRD DE-AC05-76RL01830, and by the PNNL Student Research Apprenticeship Program.

J. P. Carson is with Pacific Northwest National Laboratory, Richland, WA 99352 (corresponding author; phone: 509-371-6894; fax: 509-371-6946; e-mail: james.carson@pnnl.gov).

A. P. Kuprat, S. M. Colby, C. A. Davis, and J. T. Feo are with Pacific Northwest National Laboratory (e-mails: andrew.kuprat@pnnl.gov; sean.colby@pnnl.gov; cassi.davis@pnnl.gov; john.feo@pnnl.gov).

C. A. Basciano is with Applied Research Associates, Inc., Raleigh, NC 27615. (e-mail: cbasciano@ara.com).

K. Greene is with University of North Carolina, Chapel Hill, NC 27514. (e-mail: kgreene@unch.unc.edu)

A. Kennedy is with Cancer Centers of North Carolina, Cary, NC 27518. (e-mail: andrew.kennedy@usoncology.com)

II. Data Acquisition

A. Tissue Sample

A tissue sample was acquired from a 70 year old man with unresectable multifocal HCC who had been treated with microsphere therapy. The microspheres delivered were SirSpheres (Sirtex Medical Inc) with a mean diameter of 32μm (+/- 8μm) and containing ^{90}Y. A ~ 2cm x 2cm x 1cm block subsection of the tissue sample was then embedded in paraffin wax.

B. Tissue Sectioning and Imaging

The embedded block of tissue was serial sliced into 8μm thick sections, with 57 of the sections each spaced 200μm microns apart preserved on slides ((HistoTox Labs, Inc., Massachusetts). Slides were stained with Hematoxylin & Eosin (H&E) to highlight general histological features.

Each slide was digitized using a Hamamatsu NanoZoomer imaged at 20x scan mode (Fig. 1). This created an image at 0.460μm resolution corresponding to each slide. Total size per image was approximately 80,000 x 50,000 pixels (~4 gigapixel image). Images were initially stored in 24-bit RGB ndpi format, and then converted to Bitmap format for accessibility. Bitmap images were then downsampled to ¼ scale and stored as portable network graphic (png) images for analysis.

III. 3D Reconstruction

A. Labeling Tumor and Microspheres

An image mask representing the location of the tumor in the liver tissue was created for each tissue image. Using Photoshop, this mask was manually drawn as a green region on the tissue image by a liver pathologist. Python and Python Imaging Library were used to generate a binary mask image representing the tumor from the tumor-marked image.

A list of microsphere coordinates was created for each tissue image. A methodical visual search of the image was performed, with each detected microsphere manually marked in Photoshop with a single green pixel. Once all microspheres in an image were marked, that image was processed using a Python script to create a text file listing of microsphere coordinates. Microsphere coordinates were rapidly validated by inspecting a microsphere summary image. This summary image was constructed by a Python script that automatically cropped the tissue image around each microsphere coordinate, and then tiled the resulting

cropped images into a single image. Using this approach, we detected and verified 12,707 microspheres.

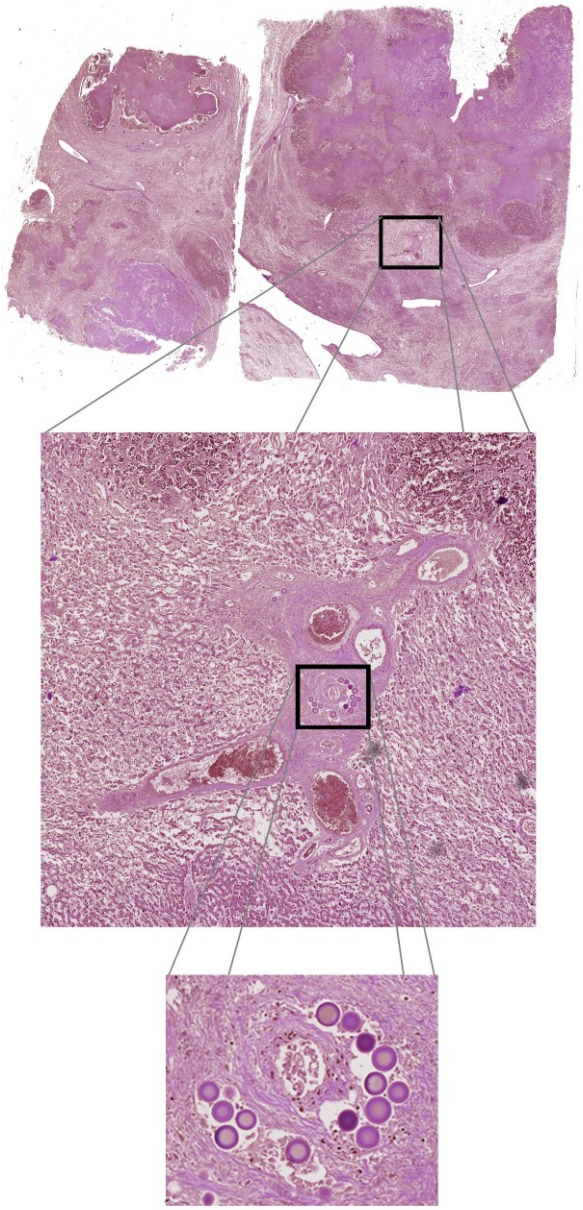

Figure 1. Digitized liver tissue sample. Location of microspheres is highlighted.

B. Pre-alignment Processing

Misalignments between adjacent images prevent direct 2D-to-3D data conversion and require appropriate image-to-image registration to restore the digitized series of 2D images representing the tissue sample to a 3D state. During serial sectioning, each tissue slice was placed on a separate slide, causing misalignment between imaged sections tissues. Addressing this type of misalignment requires rigid registration (translation and rotation) of images. In addition, generalized and localized tissue expansion was introduced as part of the standard paraffin sectioning histological technique. Removing these distortions requires unique 2D elastic transformation determined for and applied to each image.

To prepare the images for registration, images were first downsampled to $1/16^{th}$ original scale. Next, images were converted to 8-bit grayscale by subtracting the green color channel from the blue color channel followed by uniform zero-padding (5632x5632). Then, the padded grayscale images were downsampled to $1/128^{th}$ original scale.

C. Rigid Registration

During rigid alignment, the translation and rotation necessary to align the tissue section images was calculated [5]. This calculation was performed using an iterative multi-scale pyramid approach, with resolution increasing each step in order to further refine the optimal translation and rotation to align images [6]. The difference between two images was based upon the sum squared difference of overlapping pixel intensities. The difference score for an image to be translated was averaged over up to three prior rigid-aligned images in the image series (as available). Once the appropriate image-to-image rotations and translations were calculated, they were applied to the tissue images, the tumor masks, and the microsphere coordinates. This brought all three sets of data into 3D rigid registered co-alignment, thus addressing the registration issue caused by placing the tissue sections on different slides.

D. Elastic Registration

To address paraffin-related image distortions, we calculated and applied elastic registration to ensure optimum continuity of the image sequence. To handle deformation outliers, we calculated and averaged four warps for each image—to the two images preceding and the two images following the image in the series (when available). Warps were calculated using the bUnwarpJ plug-in [7] for the imaging software package ImageJ. Parameters applied were "Coarse" initial warp and "Fine" final warp. The bUnwarpJ image registration algorithm represents the images and deformations as cubic B-splines, and then utilizes an energy function summing the dissimilarity in both directions between the source image and target image. Optimization is by Levenberg-Marquardt. After averaging the four warps, a single transformation per image was saved as a text file for later application.

The calculated warp definitions were then converted via a bUnwarpJ macro from elastic warp definitions—which consider translations of the vertices of an affine grid overlay—to raw transformations—which contain the translational information for each pixel. Another bUnwarpJ macro applies the raw definitions to the liver tissue images and the tumor masks. The process of calculating and applying warps was automated as a Python script.

Surfaces representing the tissue block and the tumor were generated with the Digital Data Viewer software using a variant of marching cubes [8]. Visualization of the tissue block, the tumor, and the microspheres was performed in Paraview [9] (Fig. 2).

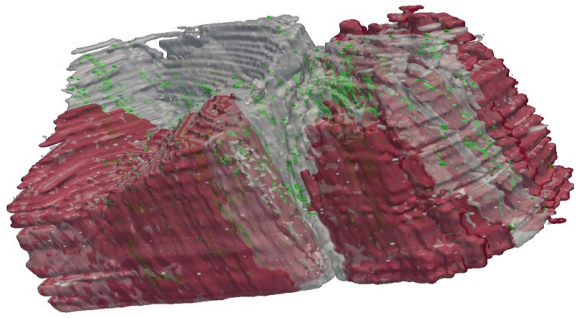

Figure 2. Reconstruction of the liver tissue block in 3D. Location of the HCC tumor is indicated by the red surface, while the tissue block is depicted in grey. Microsphere locations are indicated in green.

IV. DISTANCE CALCULATION

For each microsphere location query point **q** we needed to classify whether the point is inside or outside the volume enclosed by the tumor surface *S*, and we needed to also determine the distance between **q** and *S*. We employed the point classification algorithm outlined by Khamayseh and Kuprat [4], since it automatically provides the distance to the surface as a by-product. In [4, Algorithm 2.1], it is assumed the surface is composed of a union of smooth surface patches (e.g. triangles) and the surface does not intersect itself. It is allowed that the surfaces form two or more connected components (and hence enclose two or more volumes). In the case of disconnected components, the algorithm will return the correct inside/outside status and the distance to the nearest surface component. Further, it is assumed that the surface S is oriented and an outwards normal **n**(**p**) is defined for each point **p** in the interior of each surface patch. For the case of a piecewise linear surface, **n**(**p**) is simply the outwards normal of the unique triangle that **p** belongs to.

The idea of [4, Algorithm 2.1] is a nearest point **p** on *S* to the query point **q** is rapidly found by some algorithm and then we check the dot product (**q** - **p**)·**n**(**p**). If the dot product is positive, **q** is outside *S*; if it is negative, **q** is inside *S*. Although this is clearly true for a smooth surface, for a nonsmooth surface (such as a piecewise linear surface), a closest point **p** on *S* could be on the shared boundary of two or more patches and thus not possess a classical outwards normal **n**(**p**). In [4] the problem is resolved by defining a "synthetic normal" that is a weighted linear combination of incident patch normals, where the weight for each patch is the angle that the patch subtends at **p**. It is rigorously shown that using the angle-weighted synthetic normal, the "dot product" criterion always provides the correct result. That is (**q** - **p**)·**n**(**p**) > 0 if and only if **q** is outside *S*, even if **n**(**p**) does not classically exist, but is a synthetic, angle-weighted normal.

Following [4] we employ an skd-tree to rapidly determine a closest point **p** on *S* to each query point **q**. The skd-tree is a bounding volume hierarchy (BVH) which consists of a balanced binary tree where each leaf node corresponds to a patch (triangle) of *S* along with a minimal bounding box oriented with the Cartesian axes. Each non-leaf node consists of a minimal Cartesian bounding box containing the bounding boxes of the two children of the node. Consequently, the root node of the tree is the minimal Cartesian bounding box for the whole geometry. The construction of the skd-tree takes $O(N\log N)$ time, where N is the number of patches (triangles). In order to utilize the tree to find the closest point **p** on *S* to **q**, the root node is initially placed on a stack. Also a conservative estimate *d* of the maximum possible distance from **q** to *S* is defined as the point on the bounding box of the root node that is furthest from **q**. The algorithm proceeds by popping the node on top of the stack and (assuming it is not a leaf node) placing the children of the node back on the stack if it is possible that they possess a point that is within a distance *d* from **q**. If the popped node is a leaf, and the bounding box of the patch contains a point within a distance *d* from **q**, it is placed in a 'candidates list' of patches that possibly contain a closest point **p**. As the algorithm proceeds, the bounding boxes of popped nodes become smaller and the conservative estimate *d* can be rigorously reduced, leading to a significant reduction in the number of nodes considered. Finally, when the stack is empty, the 'candidates list' of patches that possibly contain a closest point is examined one by one and a closest point **p** is obtained from all the candidate patches. In the case that the patches are triangles, this means considering the possibilities that a potential closest point **p** is in the interior of the triangle, somewhere on the three edges of the triangle, or is one of the three vertices of the triangle. These cases are rapidly tested in $O(1)$ time.

Due to the rigorous nature of the algorithm, the closest point **p** found in the list of candidate patches is indeed a closest point on *S* to **q** as required and the 'dot product' test that is performed using the query point **q**, the closest point **p**, and the 'synthetic normal' **n**(**p**) at **p** will be rigorously correct. Moreover, usage of the skd-tree will in the best case cut the time to process a query point **q** to $O(\log N)$, where N is the number of patches (triangles) in *S*.

Prior to distance calculation, volume conserving smoothing [10] was applied to the tumor surface consisting of 700,000 triangular facets. We found that applying the point classification algorithm (including time to build an skd-tree for the tumor surface) for all 12707 query points **q** required approximately 15 seconds using a single Intel Xeon processor. We found that 1817 microspheres were within the tumor. A histogram representing the distribution of distances from microspheres to the tumor is shown in Fig. 3. The microspheres color-coded by distance and their relationship to the tumor surface are illustrated in Fig. 4.

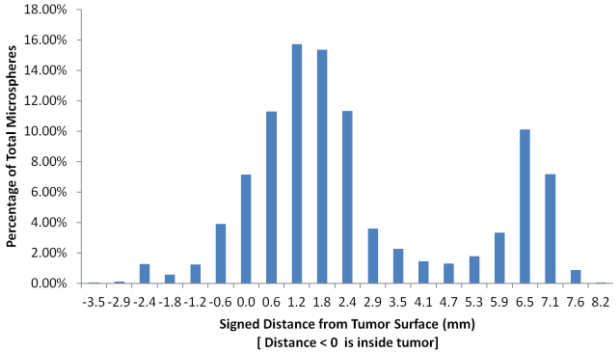

Figure 3. Distribution of distances between microspheres and tumor surfaces as represented by a histogram.

V. Conclusion

We have detailed here an approach for assessing the location of microspheres relative to a liver tumor via the process of serial sectioning, 3D reconstruction, and the calculation of distance between points and surface depicting an arbitrary volume or volumes. Due to the limited size of the tissue sample with respect to the tumor size, results are not intended as an assessment of microsphere performance, but instead as a demonstration of the method. Future directions include elastic registration at higher image resolutions, refinement of approaches for detecting microspheres, investigating the relationship of between microsphere location and blood vessels, calculating the radiation profile of the microspheres upon the tumor, and the examination of microsphere performance with different tumor types.

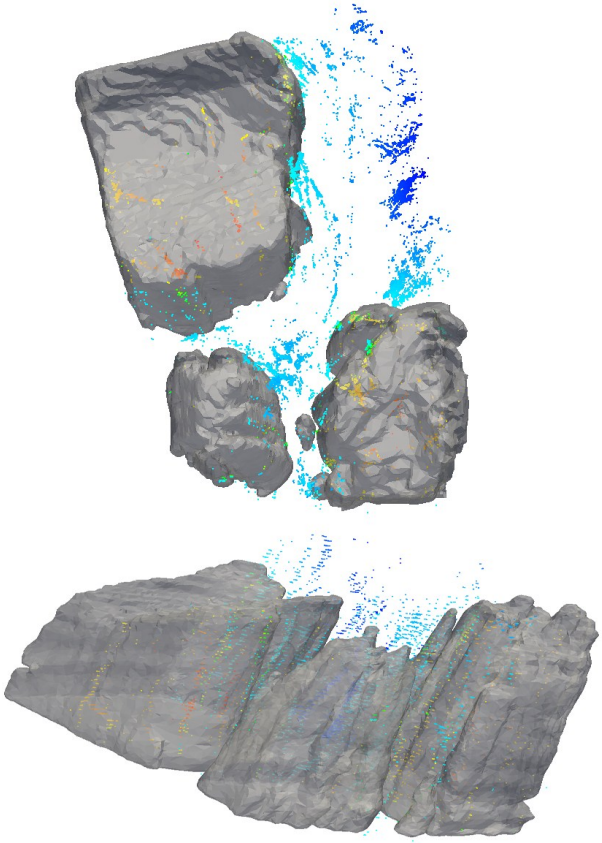

Figure 4. Relationship between the deposited microspheres and the boundary of the tumor. Two perspective views are shown (top: above the tissue block; bottom: from side of tissue block). The tumor surface is shown in grey. Microspheres are color-coded based on distance to the tumor surface (dark blue for far from tumor; light blue for near tumor; green for at tumor boundary; yellow for inside tumor near tumor surface; red for deep inside tumor).

Acknowledgment

We thank Chuck Frevert and Denny Liggitt for their assistance.

References

[1] L. Rabeneck, H.B. El-Serag, J.A. Davila, and R.S. Sandler. "Outcomes of colorectal cancer in the United States: No change in survival (1986 –1997)," *Am J Gastroenterol*, vol. 98, pp. 471– 477, February 2003.

[2] A. Kennedy and R. Salem. "Radioembolization (yttrium-90 microspheres) for primary and metastatic hepatic malignancies," *Cancer J*, vol. 16, pp. 163-75, March-April 2010.

[3] A.S. Kennedy, C. Nutting, D. Coldwell, J. Gaiser, and C. Drachenberg. "Pathologic response and microdosimetry of (90)Y microspheres in man: review of four explanted whole livers," *Int J Radiat Oncol Biol Phys*, vol. 60, pp. 1152-1563, December 2004.

[4] A.K. Khamayseh and A. P. Kuprat, "Deterministic point inclusion methods for computational applications with complex geometry", *Computational Science & Discovery*, vol. 1, 2008. Article 015004.

[5] G. Wolberg, *Digital Image Warping*. Los Alamitos, CA: IEEE Computer Society Press, 1990, pp. 47-49.

[6] S. Kovacic and R Bajcsy, "Multiscale/Multiresolution Representations," in *Brain Warping*, A. Toga, Ed. San Diego, CA: Academic Press, 1999, pp. 44-66.

[7] I. Arganda-Carreras, C. Ó. Sánchez Sorzano, R. Marabini, J. M. Carazo, C. Ortiz-de Solorzano, and J. Kybic, "Consistent and elastic registration of histological sections using vector-spline regularization," *Computer Vision Approaches to Medical Image Analysis, ser. Lecture Notes in Computer Science*, vol. 4241, pp. 85-95, May 2006.

[8] G. M. Treece, R.W. Prager, and A.H. Gee. "Regularized marching tetrahedra: improved iso-surface extraction," *Comp Graph*, vol. 23, pp. 583-598, 1999.

[9] K. Moreland, L. Avila, and L. A. Fisk. "Parallel Unstructured Volume Rendering in ParaView," *Visualization and Data Analysis 2007, Proceedings of SPIE-IS&T Electronic Imaging*, pp. 64950F-1–12, January 2007.

[10] A. Kuprat, A. Khamayseh, D. George, and L. Larkey. "Volume conserving smoothing for piecewise linear curves, surfaces, and triple lines," *J Comput Phys*, vol. 172, pp. 99-118, 2001.