Supplemental Text 2: Segmentation Methods

All of the MATLAB code utilized for segmentation and analysis is available at (GIT). The architectural and proliferation staining protocols have been previously established (REF). For the architectural staining, channels 1-4 correspond to the DAPI, DPPIV, GS, and DM stains, respectfully. For Proliferation staining, channel 2 is replaced with BrdU staining [Hammad, 2014]. Our segmentation methods are designed with an order in mind, first segmenting architectural features, followed by nuclei, and then cells.

***Architectural Segmentation***

The architectural segmentation works to segment out the sinusoids, bile canaliculi, and central/portal veins. First, sinusoids are segmented. Next, in the architectural staining, bile canaliculi are segmented. Then veins are segmented from the void space and categorized as central or portal. Lastly, the sinusoids and, in the case of the architectural staining, the bile canaliculi are improved by utilizing their skeletons to clean up the segmentation (Fig 1).

*Seg\_Sinu\_1*

First channel 2 and 4 (DPPIV and DM) from the architectural staining protocol are imported and background noise is removed by subtracting a 3d Gaussian filter of an x,y sigma of 50 and a z sigma of 10. Values under 0 are set to 0, and the images then are rescaled. We are able to subtract ch4 from ch2 to leave only areas that are potentially bile canaliculi. We then can subtract that image from channel 2, effectively removing the bile canaliculi from the ch2 image, while setting the minimum image value to 0. One concern is that smoothing the image will cause the sinusoids to become thicker than the staining suggests. However, smoothing helps to reduce noise within the image. Due to this, and Otsu’s method adaptive thresholding with a window of [101, 101, 11] is utilized on an unsmoothed image and an image with a smoothing 3d Gaussian filter with a sigma of 3, and the resulting sinusoidal segmentation is that which passes the threshold on both [OTSU]. Lastly, a spherical morphological closing operation with a radius of 3 voxels is utilized to close holes within the sinusoids, leaving us with the first sinusoidal estimation.

*Seg\_Sinu\_1\_Pro*

Within the architectural staining, we do not have the benefit of having 2 images that are stained for sinusoids, so must rely solely on channel 4. Ch4 is imported, and a smoothed image with a 3d Gaussian filter with a sigma of 3 is generated. This is done, similar to Seg\_Sinu\_1, to reduce noise while not enlarging the sinusoids. The background noise is removed from both images using the exact method within Seg\_Sinu\_1, the same form of adaptive thresholding is utilized, and the resulting sinusoidal segmentation from the smoothed and unsmoothed images are combined. This resulting segmentation then undergoes a spherical morphological closing operation with a radius of 5 voxels, followed by an opening operation of radius 3. A larger closing radius was needed to deal with missing voxels from segmentation, and the opening operation was used to open areas which were artificially connected from the closing operation.

*Seg\_Bile\_1*

Channel 2 is imported. Voxels corresponding to the sinusoid are reduced to background noise by setting their voxels to the image value of a 3d Gaussian filter of an x,y sigma of 50 and a z sigma of 10. This effectively removes sinusoids from the ch2 image while not leaving a void that would cause errors when utilizing adaptive thresholding. Background noise is then subtracted from a smoothed ch2 image created by a 3d Gaussian filter with a sigma of 3, and the same window adaptive thresholding is utilized to segment out the bile canaliculi. A 2d, 3 by 3 morphological closing operation was then utilized to fill holes. Here, a 2d operation was observed to produce better results due to the connectivity needs within the 3d operation to close holes and the thinness of the bile canaliculi.

*Seg\_Void*

In order to segment out central and portal veins, we first start by determining where areas of void are within the staining. To do so, we first bring in ch3 and ch4 (GS and DM), and smooth them with a Gaussian filter of sigma 3. Since GS is not spread across the entire image, we utilize a global threshold for ch3. This image is than closed with a cubed structuring element of width 10, and we remove any connected component with under 10,000 voxels. We utilize adaptive thresholding with the smoothed ch4 image and then utilize a morphological closing operation with a spherical structuring element of radius 5. We also import ch1 (DAPI), remove the background noise by subtracting a 3d Gaussian filter of sigma 20, and then utilize Otsu’s method for global thresholding.

Once this is complete, we combine the threshold ch1, ch4, and we import the bile segmentation to create a web in which we can look for large void spaces. We use a cubic morphological closing operator of width 20, effectively removing or drastically minimizing any void space in which a hepatocyte would exist, leaving us with only void spaces that could be explained by the presence of a central/portal vein or due to staining irregularities, which should have been removed prior to starting the segmentation. We then keep only connected components of the web which are in contact with the segmented GS stain, removing any errors in staining that may have been due to effects, such as the subtraction of background noise or adaptive thresholding, which can cause the center of large void spaces to be segmented out as foreground.

*Seg\_Void\_Pro*

The steps of Seg\_Void are repeated. However, the bile canaliculi are not added to the creation of the web.

*Seg\_Sort\_Veins*

The next step we do is to sort the segmented void components into central or portal and to eliminate void space that we believe is not a vein. To do so, we start with the void space and utilize a morphological closing operation with a cubic structuring element with a width of 10, removing any erroneous holes within the void space. Next, an opening operation with the same structuring element is utilized, removing erroneous void space. From here, we erode the void spaces with a cubic structuring element with a width of 5 and set a minimum area of the remaining space to 75,000 voxels. By eroding prior to thresholding the void space to a minimal size, we remove void space that is thin, some of which could be explained by microarchitecture features, such as bile ducts, which we did not attempt to segment within our process. The remaining void space, which we now are relatively confident is a vein, is dilated by a cubic structuring element with a width of 50 in order to sort into central or portal based on the intersection with the segmented GS stain from Seg\_Void/Seg\_Void\_Pro.

*Seg\_Veins*

Now we want to segment the actual area corresponding to each vein. For both portal and central veins, we start with the voxels of the thinned void space in Seg\_Sort\_Veins that was prior to the dilation needed for the sorting into central or portal veins that intersects with the sorted voxels. We then create a boundary to that vein, which corresponds to the sorted voxels that intersect with space that was not considered void space from the seg\_void/seg\_void\_pro algorithms. We then use an iterative dilation and erosion technique that allows for the thinned void space to fill out the void space boundary without filling in the space behind the boundary. To do so, we iteratively dilate the void space and then erode the void space with the boundary. Once this is complete, a spherical morphological closing operator of radius 8 followed by an opening operator of the same structuring element is completed to smooth out the edges of the shape.

*Seg\_Sinu\_2*

Here, we utilize the sinusoidal skeleton to clean up the sinusoid segmentation. We first start by combining the thresheld, smoothed image from Seg\_Sinu\_1 with the central and portal masks. Next, a morphological closing, with a spherical structuring element with radius 3, is utilized to clean up connections, and then holes are filled. This is then skeletonized and the skeleton inside of the central/portal veins is removed. The intersection of the smoothed, thresheld image, and the skeleton dilated with a radius of 4 is combined with the intersection of the first estimated sinusoidal segmentation from Seg\_Sinu\_1 and the skeleton dilated with a radius of 16. This is done because the endothelial cells that are stained make up the wall of the sinusoids. This means that the inside of the sinusoids will not be segmented on the unsmoothed image, but will be within the smoothed image. By combining the two, we are able to minimize the gap needed to be filled by morphological operators, allowing us to best maintain the actual shape of the sinusoid. We then are able to use a small morphological closing operator followed by an opening operator with a spherical structuring element of radius 3. Any sinusoidal connected component with under 1000 voxels is discarded. Lastly, the connections of the sinusoids with the central and portal veins is cleaned up by adding the voxels that correspond to a closing operation with a spherical structuring element of radius 5 at the connection points between the sinusoids and each vein to that vein’s segmentation.

*Seg\_Bile\_2*

Any intersecting voxels corresponding to central veins, portal veins, and sinusoids are removed from the segmented bile canaliculi from Seg\_Bile\_1. The bile canaliculi is then skeletonized, dilated with a spherical structuring element of radius 2, followed by a closing morphological operator with a spherical structuring element of radius 3.

***Nuclei Segmentation***

To segment the nuclei, first ch3 (GS) was subtracted from ch1 (DAPI). This is because there was interference between the stains. This maintained the staining of the nuclei, as GS does not stain the nuclei within the cells. The images were then rescaled, and a laplacian blob detector was utilized with sigma values between 8 and 14 voxels with a step size of 1 (REFS) to find nuclei. This range effectively removed nuclei that were more than likely non-parenchymal, excluding sinusoidal endothelial cell nuclei. Nuclei that intersected with the sinusoidal segmentation were considered sinusoidal endothelial cell nuclei. Nuclei within central/portal veins or sinusoids were removed, leaving hepatocyte nuclei.

***Cell Segmentation***

Cells are segmented exactly how they are segmented when using TiQuant (Friebel). Here, a distance transformation is utilized to find the distance of each voxel from its closest nuclei, *Ci*, while another distance transformation is utilized to find the distance to the nearest sinusoid, bile canaliculi, or central/portal vein, *Cs (*Maurer*)*. A distance is used that creates the largest distance value the further you are from the nuclei and the closer you are to sinusoids, bile canaliculi, or a vein, while Beta is a term used to determine the influence each distance gives:

We follow Friebel et. al. in using a Beta value of 0.1. This distance transformation is inversed, and normalized. Following that, an H-minima transformation is utilized to remove valleys within the distances of less than 0.05 (Soille), and a watershed transformation is utilized to split cells (Beare).

Hammad, S., Hoehme, S., Friebel, A., Von Recklinghausen, I., Othman, A., Begher-Tibbe, B., ... & Golka, K. (2014). Protocols for staining of bile canalicular and sinusoidal networks of human, mouse and pig livers, three-dimensional reconstruction and quantification of tissue microarchitecture by image processing and analysis. *Archives of toxicology*, *88*(5), 1161-1183

Friebel, A., J. Neitsch, T. Johann, S. Hammad, J. Hengstler, D. Drasdo and

% S. Hoehme (2015). TiQuant: Software for tissue analysis, quantification and

% surface reconstruction.

% Signed Maurer Distance Transformation

% Maurer, C.R. et al. (2003) A Linear Time Algorithm for Computing

% Exact Euclidean Distance Transforms of Binary Images in Arbitrary

% Dimensions, IEEE - Transactions on Pattern Analysis and Machine

% Intelligence, 25(2): 265-270.

Soille, P. Morphological Image Analysis: Principles and Applications. Springer-Verlag, 1999, pp. 170-171.

% watershed

% Beare, R and Lehmann G. (2006) The watershed transform in ITK

% discussion and new developments, http://hdl.handle.net/1926/202.