**Information sheet for raw MRI and DTI data**

**Manuscript title:** *Regional variation in lateral and medial gastrocnemius muscle fibre lengths obtained from diffusion tensor imaging*

**Manuscript DOI :**

**Authors:** Jeroen Aeles, Bart Bolsterlee, Nicole Y. Kelp, Taylor J.M. Dick, François Hug

**Corresponding author:** Jeroen Aeles, Jeroen.aeles@univ-nantes.fr

**Summary**

The following data set contains the raw MRI (T1-weighted) and DTI scans of the dominant lower leg of 32 adults (males, females, young adults, older adults). These data were used to analyse the regional variation in muscle fibre lengths in the medial and lateral gastrocnemius muscles. The T1-weighted scans were used to segment the muscles and the DTI scans were used to reconstruct the muscle fibre architecture. For the latter, a previously published pipeline was implemented (*Bolsterlee et al., 2019*[[1]](#footnote-1)). The Matlab scripts that were then used to further process and assign the reconstructed muscle fibres to specific muscle regions are also available with the data set. Below is additional information that can be used complementary to the information provided in the manuscript. The first section describes the implementation of the previously published pipeline for fibre tracking1. While this may provide some additional information to help describe the structure of the current data set, we recommend the documentation originally published for this pipeline1. The second section provides additional information to the reasoning and implementation of further analyses to assess the regional muscle fibre lengths from the data. Please cite the appropriate sources when using any of these analysis scripts or data.

**Fibre tracking pipeline implementation (*Bolsterlee et al., 2019*)**

**Overview**

1. Prepare data and folders
2. Using the Matlab scripts
3. Convert DICOM to NIfTI files
4. DTI average and blips
5. Eddy Current Distortions correction
6. Combine top and bottom vibe images
7. Reorient segmentation file
8. Local Principal Component Analysis filtering & Diffusion tensor reconstruction
   1. Derives eigenvalues & eigenvectors
   2. Creates map
9. Create 3D triangulated surface models
   1. Muscle volume is calculated from this
   2. Muscle length is calculated from this
10. DTI fibre tracking
    1. Fascicle lengths are calculated from this
    2. Pennation angles are calculated from this
11. Potential fixes for errors

**Matlab files required for steps below**

1. */*
2. *Parent\_script\_DTI\_MIP.m*
3. *Convert\_DCM2NII\_MIP.m*
4. *DTI\_average\_and\_blips\_MIP.m*
5. *batch\_topup\_MIP.m (+ fsl\_topup\_MIP\_iOS.m)*
6. *combine\_vibes\_MIP.m*
7. *reorient\_segmentation\_MIP.m*
8. *DTI\_recon\_MIP.m*
9. *make\_masks\_and\_surfaces\_MIP.m (+ MakeSurfaceAndMasks\_MIP.m)*
10. *DTI\_fibre\_tracking.m (+ TrackFibres\_MIP.m)*

**Software required**

1. *Matlab*
2. *DSI studio version 2016\_09 works for Windows*
3. *DSI studio version 20 Nov, 2017 works for Windows*
4. *Convert3D*

**Data required**

1. *Raw DTI file (usually called ep2d\_diff\_mddw\_12)*
2. *Blip PA map (b0\_p-a)*
3. *Anatomical scans (vibe\_we\_tra\_segmentation, should have 1 bottom and 1 top file)*
4. *Corresponding segmentations (e.g. S04.nii)*

**Example legend for steps below**

Input files

*Matlab script to run*

Output files

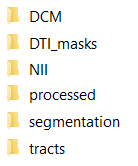
Input folder

Output folder

*\* Note that line numbers used below likely do not correspond to final uploaded scripts.*

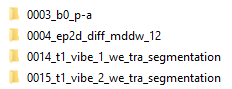
**Processing steps**

1. Prepare data and folders
2. Create folder for participant (parent folder name *‘S##’*, e.g. *S01*)
3. Copy empty folders from *‘Generic’* folder into new participant folder, alternatively create the folders below:



1. Copy data from participant to *S##/DCM*

*! Note that individual zip-folders need to be unzipped. Leave .dcm files in folders -> you can do this for multiple folders at once using 7-zip software -> select folders -> right click -> 7-zip -> extract to “\*\”*

**

**

1. Copy segmentations data from participant to *S##/segmentation*



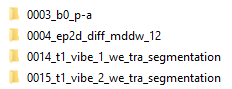
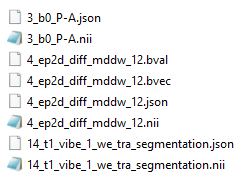


1. Using the Matlab scripts
   1. Use of the Parent script: *Parent\_script\_DTI\_MIP.m:*
      1. After installing the required software and adding to the system path (Windows – to see if this works by opening the commands window (search -> cmd) and typing in separately and without the “”: “c3d”, “C:\Users\XX\ dsi\_studio\_64\dsi\_studio.exe”, “C:\Users\ XX\dsi\_studio\_64\dsi\_studio.exe”, this should open all software for dsi\_studio and read something like “PICSL convert3d tool” for c3d, if you see this it should work), locate the software folders & add the respective paths to the Parent script l6-8.   
         Next, specify which subjects you want to analyse and for which side on l12-13, e.g. if you want to analyse both sides for subject 1 and 2 but no sides for subject 3 and only the right side for subject 4, you should fill in: [01 02] on l12 and [01 02 04] on l13 (note that the folder names in your directories should be e.g. S01 and S04, but you do not need to amend the “S” in Matlab, this is done automatically.  
         Specify on l17 which muscles you would like to analyse (this cannot be chosen individually for different subjects) by choosing a number between 1-7 (see the notes in comment on l16 for the legend).

You are now ready to hit the “Run” button and start the analyses. The scripts will only work if everything is located in the correct folders and no input is missing for any of the muscles or subjects. You can find a list of the required inputs to start the analyses on page 1 of this manual.

1. Convert DICOM to NIfTI files
2. Run *Convert\_DCM2NII.m*

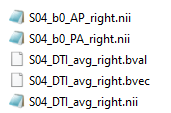
*Convert\_DCM2NII.m*



1. DTI average and blips
2. Run *DTI\_average\_and\_blips\_MIP.m*

! Note that the filenames need to be EXACT (e.g. 3\_b0\_P-A-L.json will not work (A\_L at end will work, script looks for exact match of b0\_P-A\_L match)

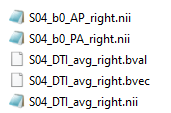
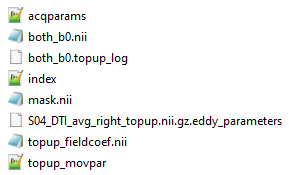
*DTI\_average\_and\_blips\_MIP.m*



1. (FSL topup) Eddy Current Distortions correction
2. Run *batch\_topup\_MIP.m*

Note: to run this on Mac OS, you need to open Matlab through the terminal (go to Matlab folder/bin then ./Matlab) otherwise it cannot use the fslmerge.

*batch\_topup\_MIP.m*

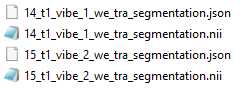


1. Combine top and bottom vibe images

! Note that you may need to adjust the filenames on l19-20 in the script below.

1. Run *combine\_vibes\_MIP.m*

*combine\_vibes\_MIP.m*



1. [Optional step(?)] Reorient segmentation file to same format as vibe\_combined.nii.gz

! Only required when 2 scans have different coordinate systems from different software

Note that the first input file should be located in the folder *‘\segmentation’* whereas the second input file is saved in *‘\processed’* in step 5.

Also note that both files need to have the exact filenames as shown below. (SXX.nii)

1. Run *reorient\_segmentation\_MIP.m*

*reorient\_segmentation\_MIP.m*



1. Local Principal Component Analysis filtering & Diffusion tensor reconstruction

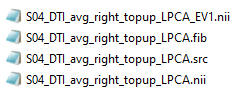
! You need to install the 2016\_09 version of dsi\_studio for this step and add it to the

Windows system path, then you need to specify the location of the executable

(dsi\_studio.exe for windows) on l6 of the script *DTI\_recon\_MIP.m*

1. Run *DTI\_recon\_MIP.m*

*DTI\_recon\_MIP.m*



1. Create 3D triangulated surface models
2. Run *make\_masks\_and\_surfaces\_MIP.m*

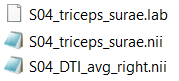
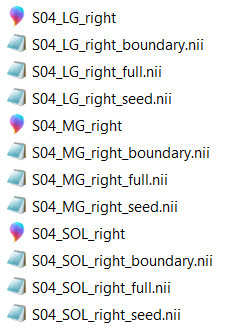
! Note that this requires the *MakeSurfaceAndMasks\_MIP.m* script in the

muscle-dti-toolbox folder with all functions

! Note that the first two input files should be located in the folder *‘\segmentation’*

whereas the third input file is located in *‘\processed’*.

*make\_masks\_and\_surfaces\_MIP.m*



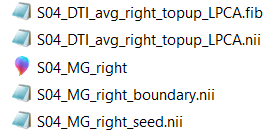
1. DTI fibre tracking

! You need to install the Nov 20, 2017 version of dsi\_studio (2019 induces bugs!) for this step & add it to the Windows system path, then you need to specify the location of the executable (dsi\_studio.exe for windows) on l8 of the script

*Parent\_script\_DTI\_MIP.m*

1. Run *DTI\_fibre\_tracking\_MIP.m*

*DTI\_fibre\_tracking\_MIP.m*



1. Potential fixes for errors (may be outdated)
   1. Make sure all software are the versions mentioned in this document (where applicable)
   2. Make sure all software is added to the (Windows) system path
   3. Make sure you have 2 different versions of DSI\_studio installed and that the directories are linked to these folders correctly (in parent script)
   4. We noticed that the version of DSI\_studio is very important. For example, the latest 2019 version induced bugs in the Fibre Tracking step, as it added many duplicate tracts.
   5. For issues with the *MakeSurfaceAndMasks\_MIP* script, a potential fix is to:
      1. Install another version of Convert3D software
      2. Install the Convert3D software in a different folder (e.g. documents)
   6. For step 7 – files need to have EXACT name – “SXX.nii” (e.g. not S10\_v2.nii). This may also be a fix for other steps, some scripts look for exact filenames. If you are stuck, compare the input filenames from a subject that works with the ones that give errors.
   7. If there is a combined vibe already, you can try to convert it to the correct file type (.nii.gz):
      1. open data in ITK snap using FILE-> OPEN MAIN IMAGE
      2. -hen go SAVE IMAGE (choose MAIN IMAGE) and select where and the type of file you want save image (as NIFTII format)
   8. If an error occurs “DTItracts does not contain a field tracts\_xyz,…” line 88 in CalcArchitecture, line 115 in DTI\_fibre\_tracking\_MIP, check if all the data is used for the same (left or right) leg! You can easily check this by using ITK Snap and opening main image: the vibe\_combined file. Than in ITK Snap -> file -> add another image -> open the DTI file (NII folder -> *ep2d\_diff\_mddw\_12* file), you can also add segmentation (e.g. S01.nii.gz in folder segmentation). To see if this is all on the correct and same leg.
   9. If the segmentations are labelled incorrectly (correct = 1,2,3 for right leg; 4,5,6 for left leg), you can re-name the labels in ITK-SNAP -> load main image (vibe\_combined), then segmentation (e.g. S14\_triceps\_surae.nii.gz), then click on “label editor” (looks like a painting pallet) and re-label. ***Note*** *that when you run the “reorient\_segmentation” script, this may again undo your re-labelling, so this should be done before re-labelling.* Also check that the “lab file” contains the correct numbers.

**References**

Bolsterlee, B.; D'Souza, A.; Herbert, R. D. (2019): Reliability and robustness of muscle architecture measurements obtained using diffusion tensor imaging with anatomically constrained tractography. In *Journal of biomechanics* 86, pp. 71–78. DOI: 10.1016/j.jbiomech.2019.01.043.

Bolsterlee, B.; Veeger, H. E. J. D.; van der Helm, F. C. T.; Gandevia, S. C.; Herbert, R. D. (2015): Comparison of measurements of medial gastrocnemius architectural parameters from ultrasound and diffusion tensor images. In *Journal of biomechanics* 48 (6), pp. 1133–1140. DOI: 10.1016/j.jbiomech.2015.01.012.

**Additional information for the assessment of regional muscle fibre lengths**

**Data reduction for removing potential noise**

Fibres that were extrapolated for more than 30% of their total length were excluded form further analyses.

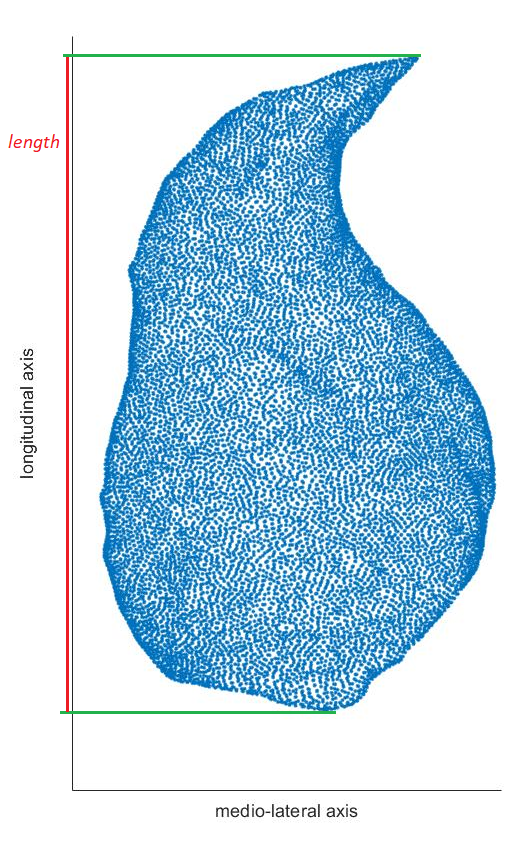
**Transforming to local muscle coordinate system**

The muscle and fibre coordinates were transformed from the original scanner coordinate system to a local muscle coordinate system, using a Principal Component Analysis. The coordinates were used as input in the PCA and the output provided the transformation matrix. The first principal component (PC) corresponds to the longitudinal axis of the muscle, the second PC to the medio-lateral axis, and the third PC to the anterior-posterior axis. These newly transformed values were used for further analyses.

**Muscle dimensions**

Muscle dimensions were calculated in order to explore potential correlations between the dimensions of the muscle and its architecture, volume, etc.

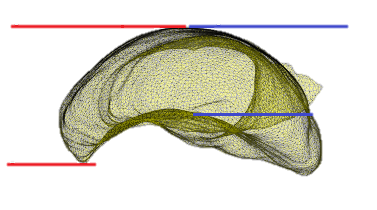
* Muscle length was defined in the most common and simplistic way: as the difference between the maximal and minimal coordinate along the muscle’s longitudinal axis (Fig. T1, MG example, red line equals the length).



*Figure T1:*

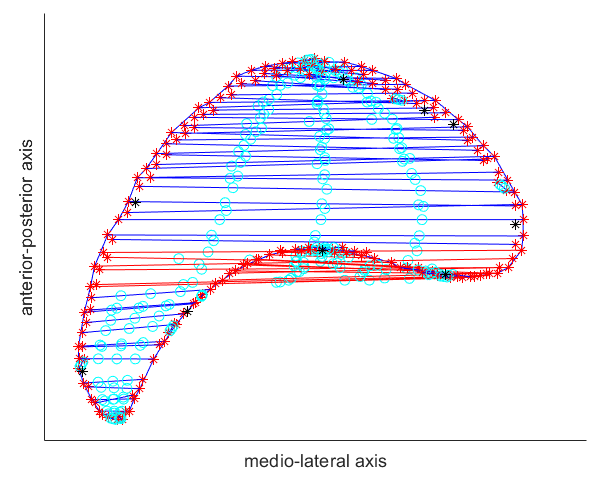
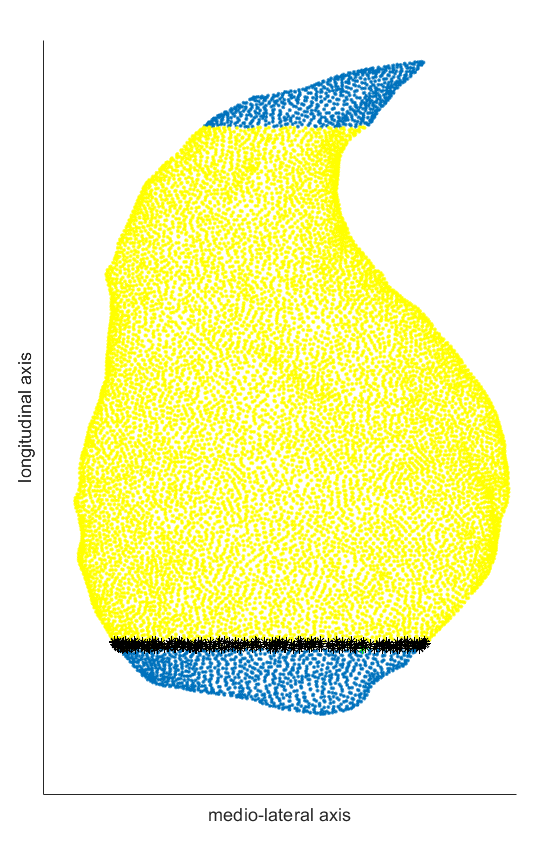
Due to the complex shape of the muscle and seemingly large inter-individual (and inter-muscle) differences in the shape, a more complex but more accurate approach was taken to determine the muscle’s width and thickness. The example below (Fig. T2) shows the reasoning on a cross-sectional image of MG. Using the same method as was done for muscle length, the thickness would be calculated as the distance between the two red lines leading to an overestimation. It is more accurate to calculate it as the distance between the two blue lines. The calculated parameter is then termed “maximal width/thickness” of the muscle.

This is computationally more complex but can be done by going through each “slice” such as the one shown in Fig. T3 below, and finding the distance between two “paired points” on opposite sides of the muscle. The description below is fist to determine the muscle width.



*Figure T2: Medio-lateral (x-axis) – anterior-posterior (y-axis) view of the muscle.*

This was done for slices of thickness 3mm along the longitudinal axis (black band in Fig. T3). *(Note that for programming reasons, the most proximal and most distal 10% of the muscle’s length were excluded from the analysis).* All points in this band were then plotted in the plane as shown on Fig. T3. A random point was selected in this band and a corresponding “opposite” point that lay within 3mm (longitudinal axis) and 1mm (anterior-posterior axis) range was detected. Often, multiple points laid within this range, if this was the case, the distance along the medio-lateral axis (i.e. muscle width) was calculated and the highest value was selected. Then these two “paired” points were removed from analysis and their coordinates as well as the corresponding muscle width (calculated as the distance between them, i.e. blue lines on Fig. T3) were stored. This process was repeated for all points in the slice. Another restriction had to be implemented because of the shape of the muscle, i.e. the distance measure had to be entirely “inside the muscle” (e.g. exclude red lines in Fig. T3).



*Figure T3: Start of muscle width calculation process. The yellow dots (left image) represent the middle 80% of the area’s length in which the maximal width was searched for. The black band is the first “slice” in which the width was determined. An anterior-posterior – medio-lateral view of this slice is shown in the image on the right. The ‘\*’ symbols show the datapoints of the muscle boundary, and the width was searched between these points along the medio-lateral axis with the restrictions mentioned above in place. The blue lines represent valid muscle width values, whereas the red lines represent solutions which violate the restriction of not lying entirely inside the muscle. The cyan circles were used to enforce this restriction check and can be ignored. The values of muscle width (i.e. length of lines) of all the blue lines were stored together with their corresponding “paired” points (‘\*’).*

This process was repeated for all slices along the longitudinal axis, storing 1 value of “maximal width” for each slice, and finally the maximal width value for the entire muscle was determined (max value from all slices). A few more examples of the slices and their calculation solutions are shown below as well as the final location of the muscle’s maximal width (blue line). Note that because of the reasons mentioned above, this width is not equal to the distance between the maximal and minimal values along the medio-lateral axis (due to some of these boundary values having different values in the third (missing on these figures) dimension.

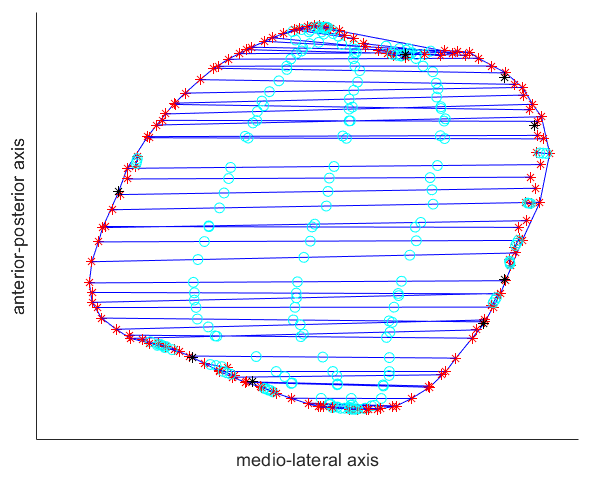
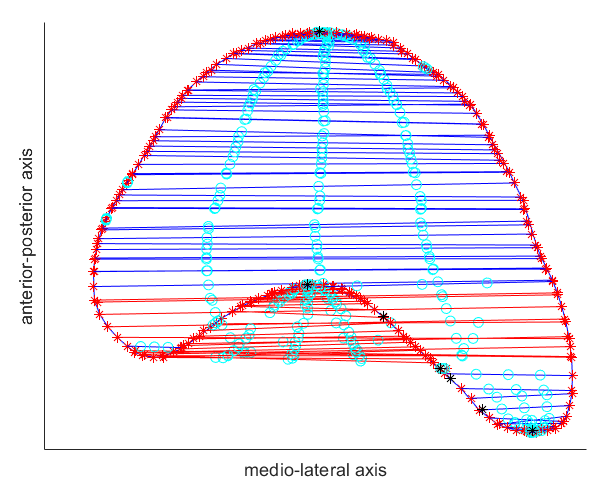
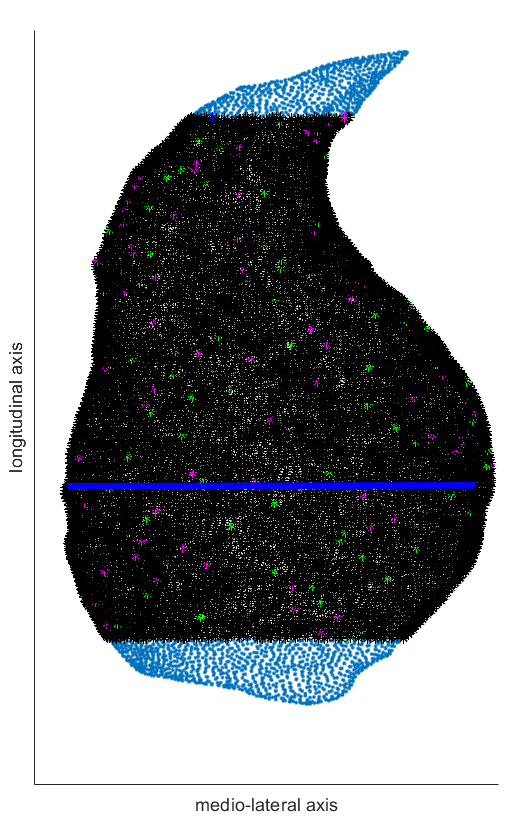


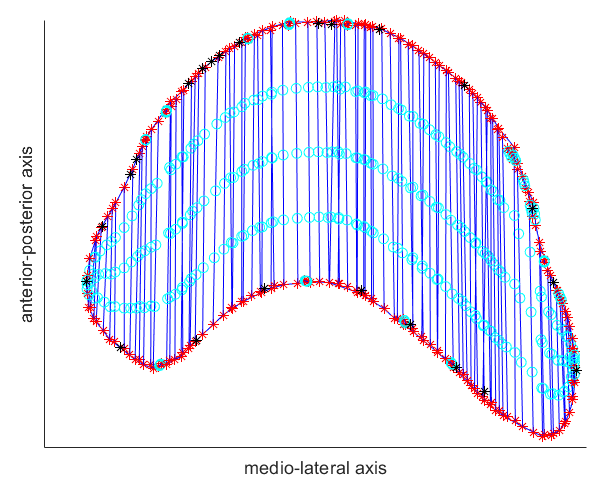
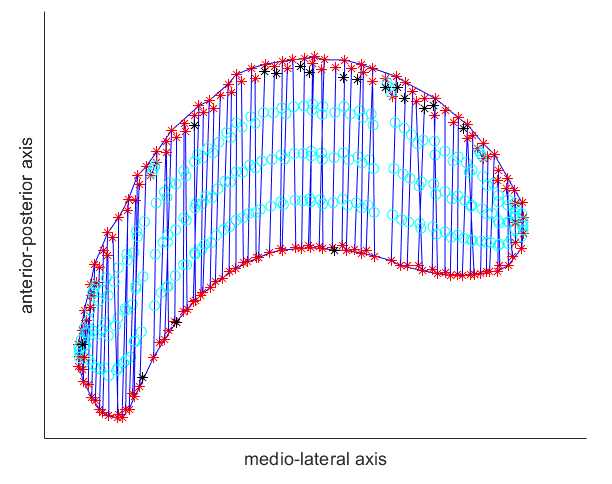
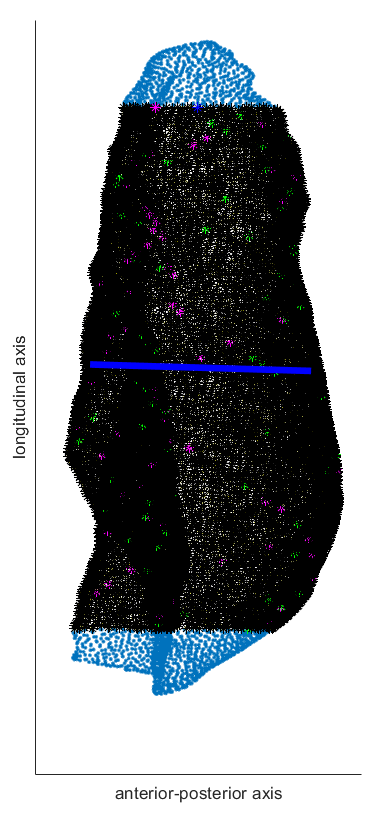
Figure T4: Example slices along the muscle’s longitudinal axis. See caption Fig. T3 for details.



*Figure T5: Final result with the maximal muscle width shown as the thick blue line. The maximal width of this example was 87 mm (versus 92 mm if using less accurate maximal and minimal value method).*

Muscle thickness was determined using the same approach, by finding the maximal thickness along the anterior-posterior axis. Example images are shown in Figure T6.

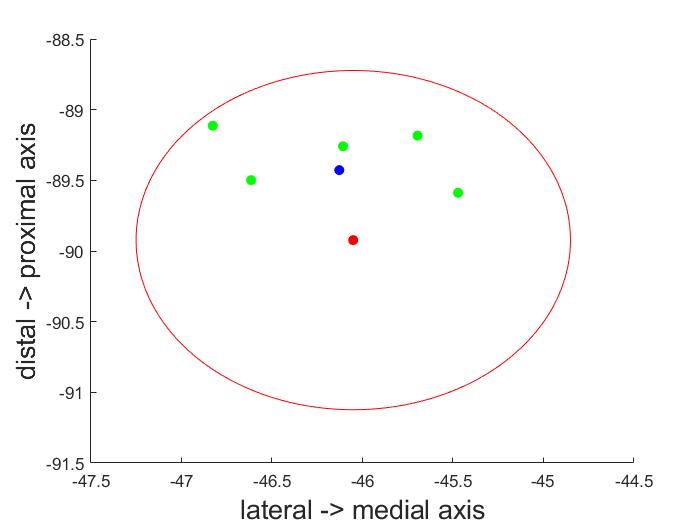
* Maximal muscle thickness was defined as the largest value obtained from thickness measures on cross-sectional slices along the longitudinal axis of the muscle.



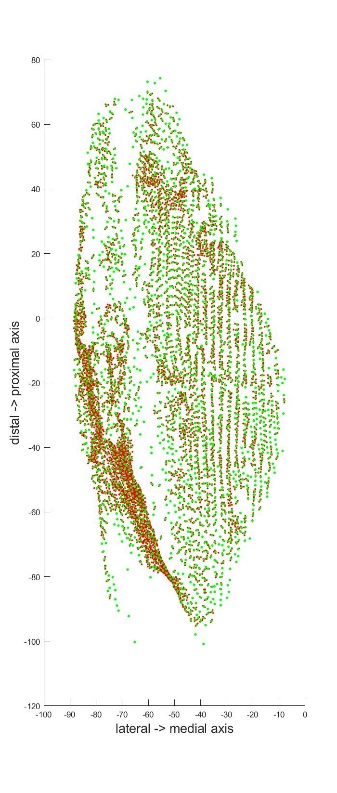
*Figure T6: Example images for determining the maximal muscle thickness. Details described in Fig. T3. The maximal thickness of this example was 41 mm (versus 50 mm if using less accurate maximal and minimal value method).*

**Fibre grouping**

In order to compare regional anatomical variation between individuals and between muscles, and to evaluate the reliability of such potential anatomical variation, fibres within a specified area were grouped together. Fibre midpoints (i.e. the middle of the straight line between the attachment point on the deep and superficial aponeuroses) were used for this. Two approaches were initially taken: one in which the fibres within a certain radius around semi-randomly (from distal to proximal) fibre midpoints were grouped and averaged. The concern that was raised with this method was that the location of the circles and thus in which areas of the muscles the fibres were grouped would vary between individuals and between muscles. Another issue was that the new location of the “average fibre” was determined by calculating the average location between all fibres in the group. This location therefore depends on which fibres are included in the group and also how many “fibres” were picked up in the measurement. An example of the method is shown in the two figures below.

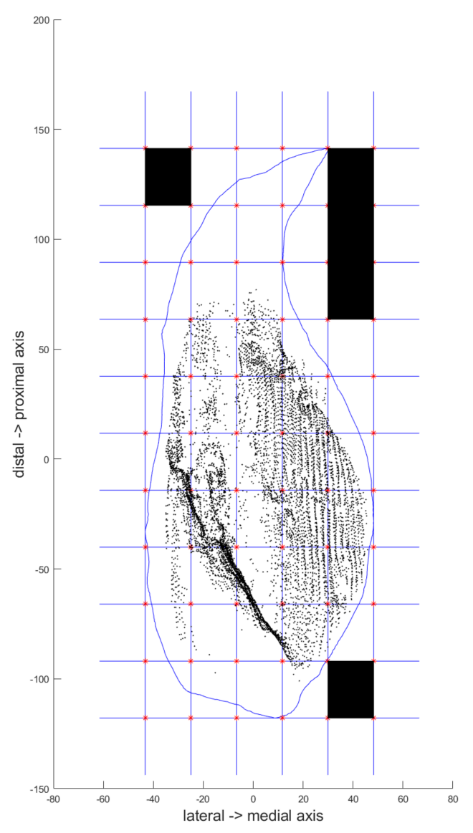


*Figure G1: red and green dots represent midpoints. The red point was selected as the fibre group’s centre. Then the median fibre length was calculated between the red and green midpoints, as well as their mean location (in this plane), represented as the blue dot.*

**

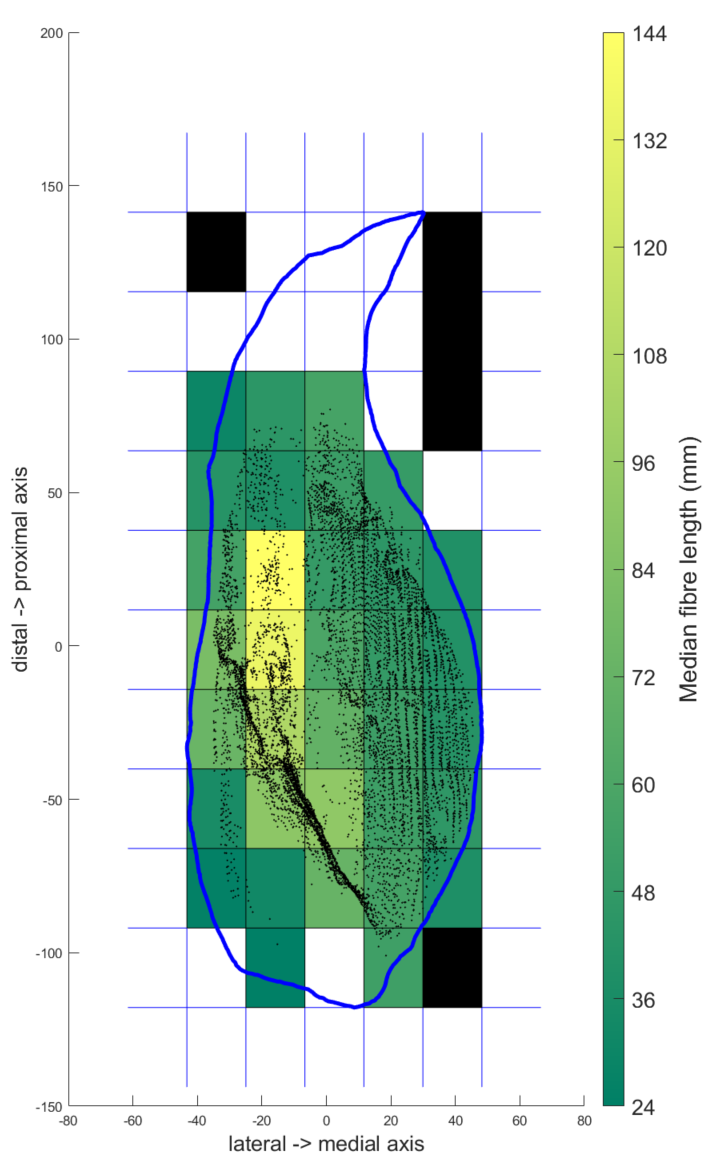
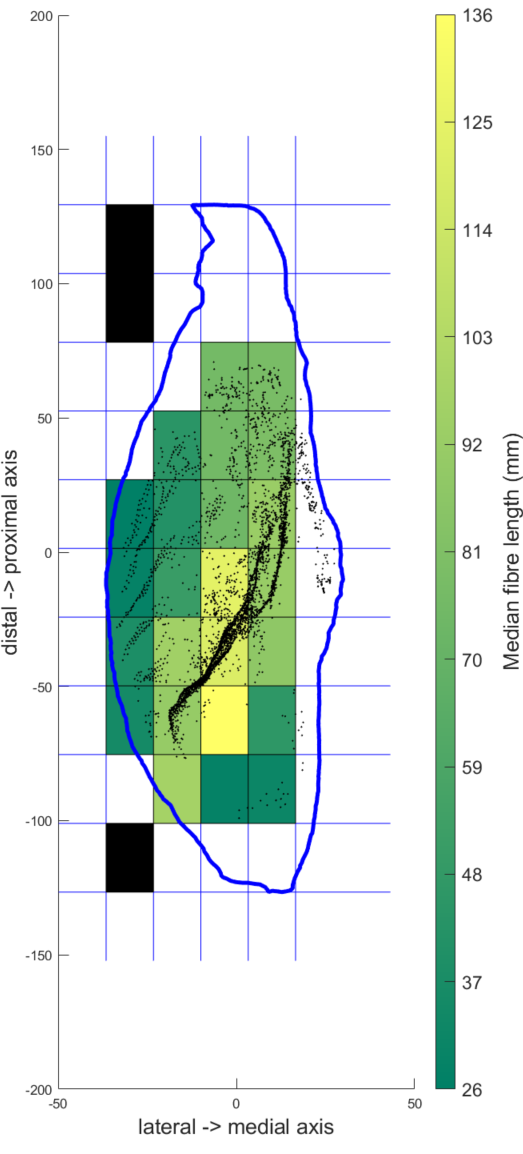
*Figure G2: Result of using this method, red dots are the original midpoints, green dots represent the mean locations of the new “average fibres”, for which the median fibre length was used as “new, average length” for the group.*

The second approach sought to remove these issues by (almost-)systematically dividing the muscle in areas and grouping the fibres within each area. For this approach, I created a grid of rectangles with 10% of the muscle’s length for rectangle height and 20% of the muscle’s width for rectangle width, starting and ending at the extreme sides of the muscle’s boundaries as shown in figure G3. *Note that the “outside muscle” boundaries were used here and not the actual width as calculated with the methods mentioned above (would be pointless).* Then the midpoints of the fibres were overlayed on this image and the fibres within each rectangle were calculated using their coordinates (red markers below). The black rectangles represent rectangles that have no muscle inside them and were thus excluded early from analysis.



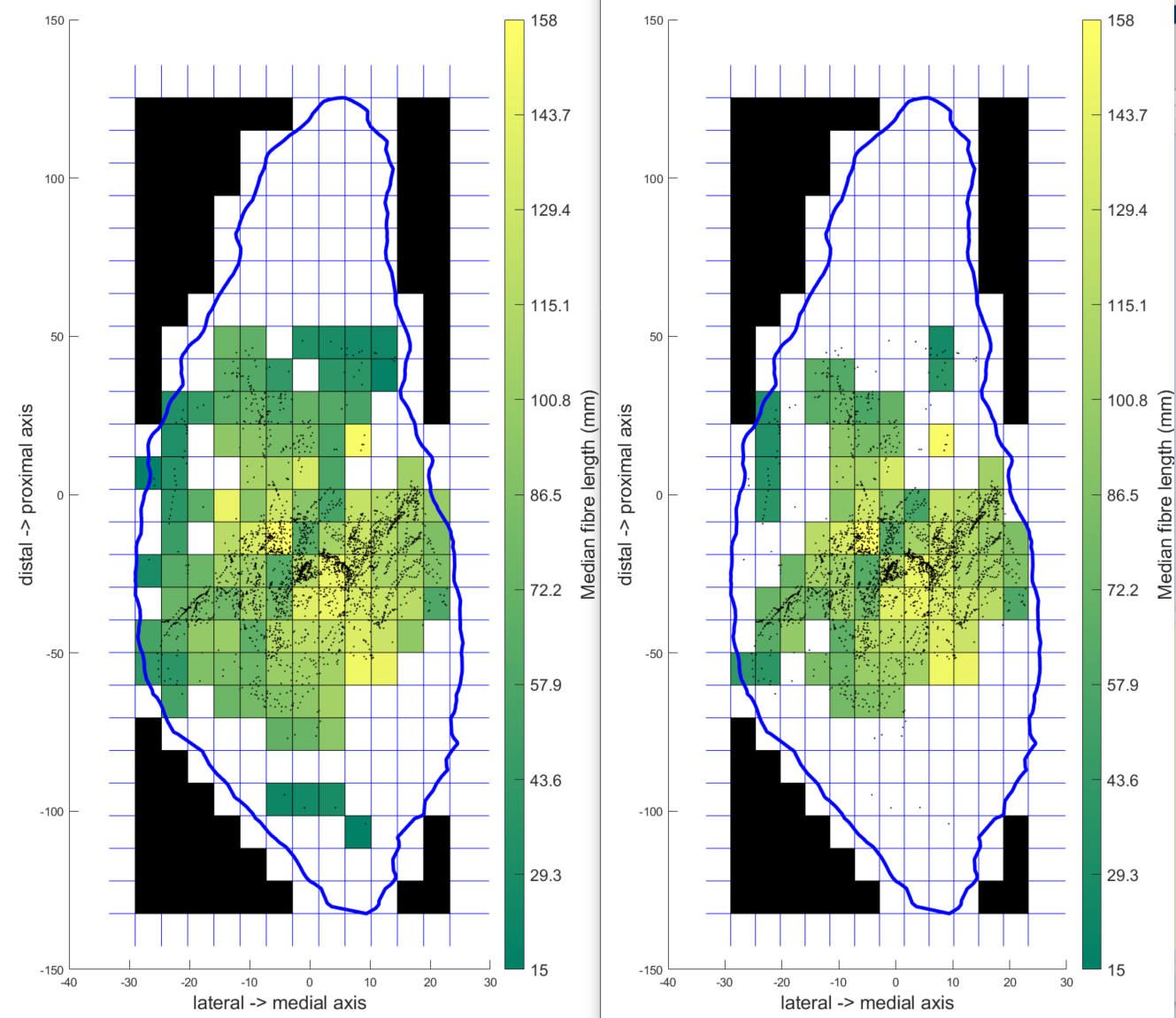
*Figure G3: overlayed grid on the muscle boundaries. Grid size was defined as a percentage of the muscle width and length. Black dots represent midpoints. Black rectangles were excluded from the analysis as they had no muscle inside.*

At this point, the median fibre length (and pennation angle) per rectangle were calculated using the values from all the midpoints inside each respective rectangle. As a first quick visualisation of the results, I used a colormap on the grid, where the colour intensity represents the median fibre length (Figure G4).



*Figure G4: Similar as G3, but with the rectangles colour-coded and for both MG (right) and LG (left) of the same participant (left leg). Note that the photos are stretched randomly so they do not represent actual muscle dimensions. The colour intensity represents the median fibre length of the midpoints within each rectangle. White rectangles have 0 midpoints inside them.*

I also added an extra restriction, namely that at least 3 fibres need to be in the grid for it to be highlighted in the colouring. The effect is visible on Fig. G5 for an example of the LG muscle of a random subject. For this case, smaller grids were taken, and the grids that have less than 3 fibres in them, are left white. The value of 3 is chosen arbitrarily. This was only used for visualisation and not further used.



*Figure G5: effect of removing grids with less than 3 midpoints per grid. Left figure shows the original method, the right figure ignored these midpoints (left them uncoloured).*

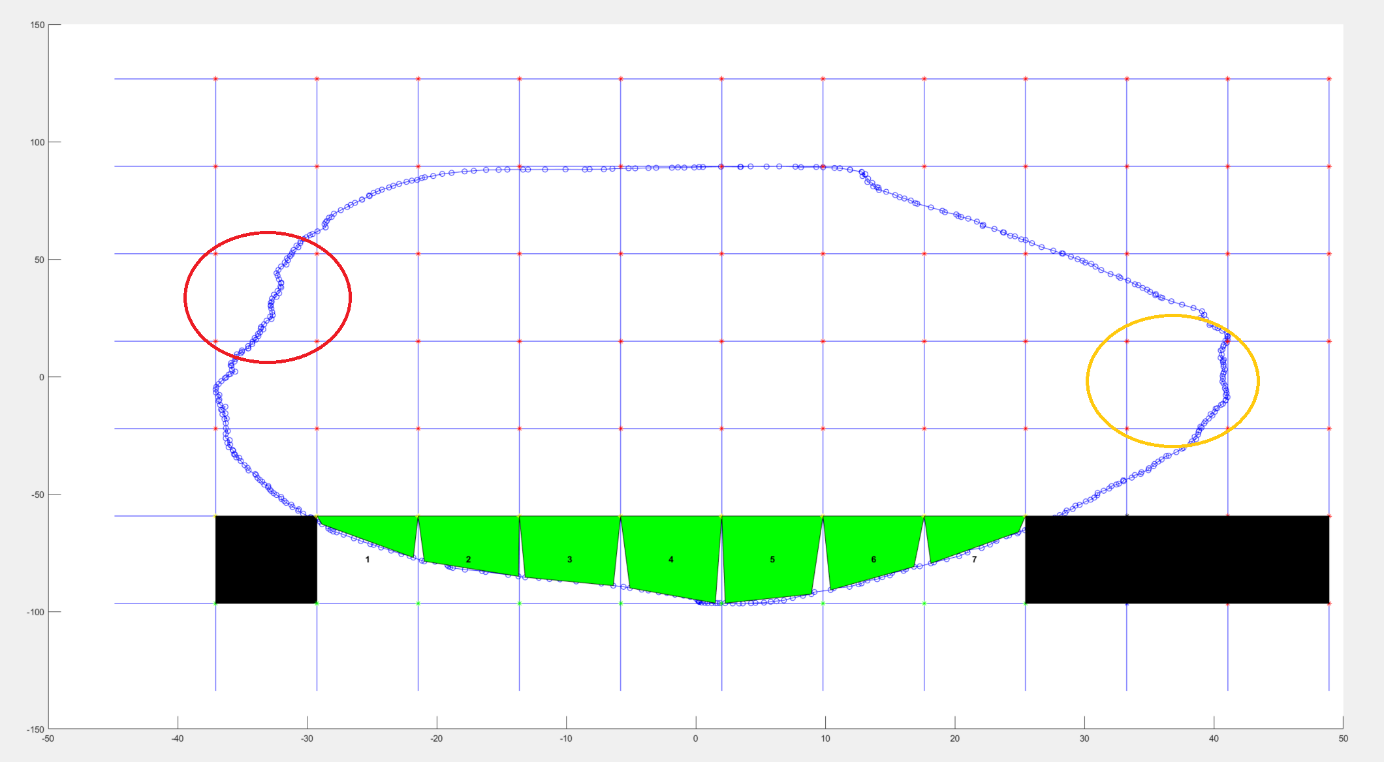
**Rectangle matching between scans (Reliability)**

Below is an example of the grid comparison for size T rectangles (Fig. G7). It was important to accurately match each rectangle from scan 1 with the corresponding rectangle from scan 2. This was very challenging for smaller rectangle sizes because of the highly “random” muscle shape and small differences between segmentations from different scans. A combination of methods was applied which resulted in an incomplete solution that combined automatic and manual processing.

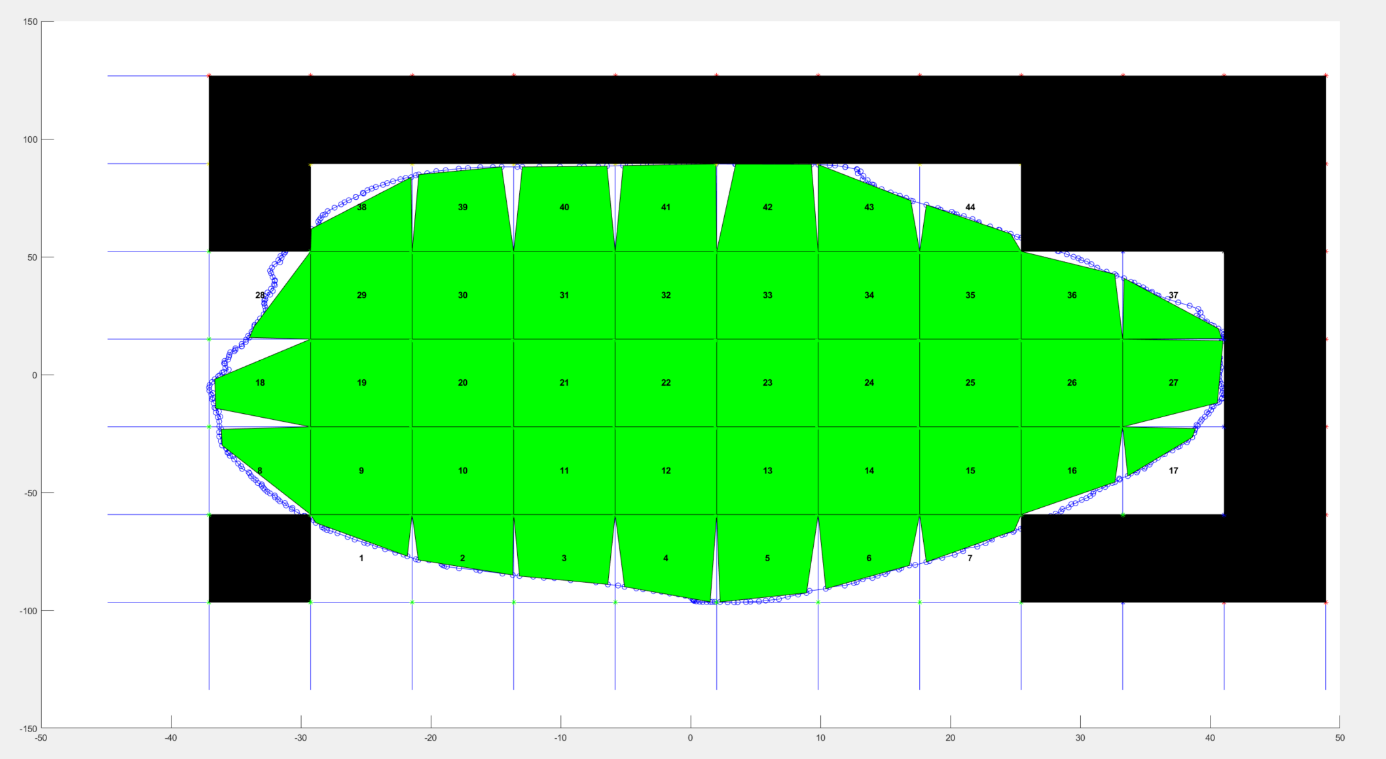
In a first step, the muscle boundary data points were assigned to each rectangle, together with the coordinates of the rectangle corners. Then, a combination of if-loops and for-loops were implemented that aimed to identify the “type of shape” the muscle boundary made within the rectangle in order to set the best criteria for calculating the “corners” that defined “within muscle area”. An example is shown in Fig G8.



*Figure G7: Data is from 2x2x2 mm seed spacing, left figure is for scan 1, right figure for scan 2. Coloured dots show midpoints assigned to a specific rectangle. The light green fills were for a method implemented to match rectangles between grids, namely to detect only rectangles with more than X amount of “within-muscle” surface area in a rectangle. The value for X was varied until an appropriate value was used. The middle figure shows the rectangles that had midpoints in both scan 1 and scan 2, which were then further used for analyses. The colour scale shows the median fibre length for scan 2. Note that the vertical axis label is flipped here…*

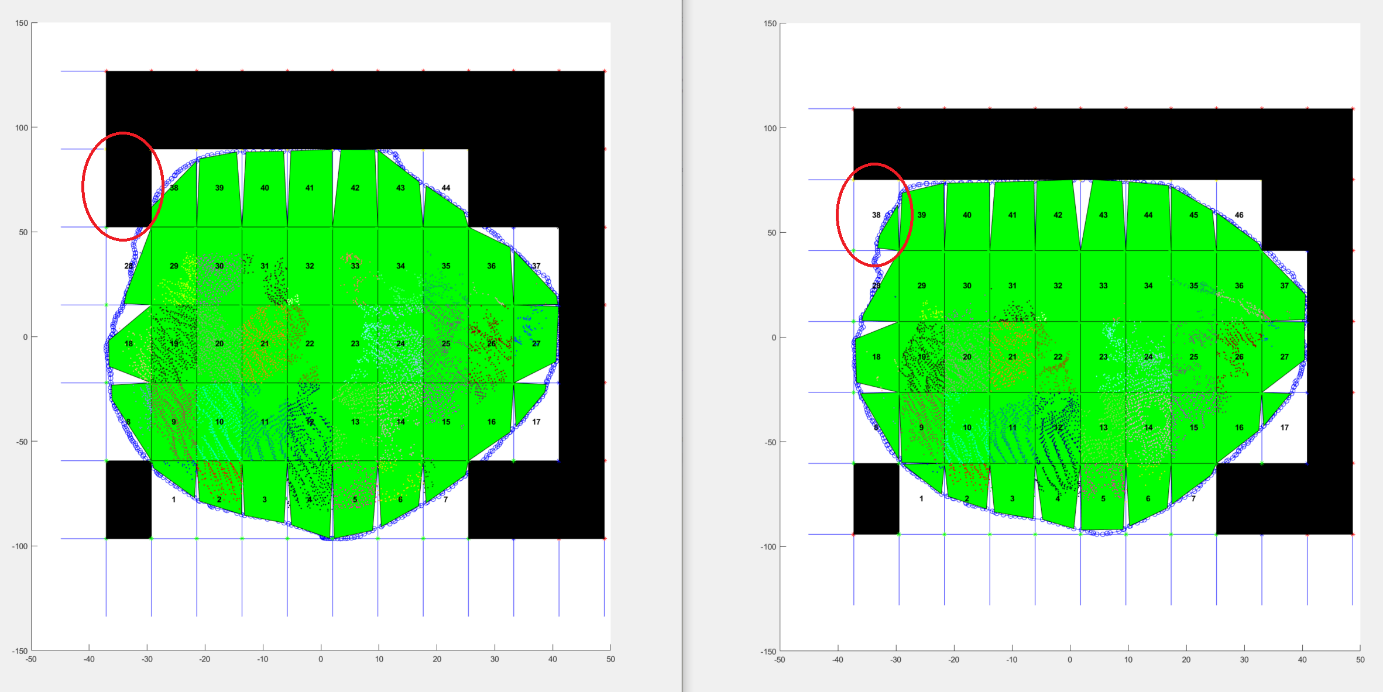


*Figure G8: example of why the automated surface area of the “within muscle area” per rectangle was challenging. Note the difference between the “borders” in the red and orange circles. The borders need to be defined in order to use the ‘polyarea’ function in Matlab. As shown in Fig G9, this approximation worked well for some rectangles bus less for others. Multiple adjustments and criteria were implemented as well as different mathematical approaches to the problem. They consistently imrproved the approximation for a certain number of rectangles but then made it worse for other rectangles.*



*Figure G9: results example for the automated approximation of “within muscle area” implementation. The figure clearly shows how the approximation worked well for some rectangles, while it was further off in other rectangles. These surface areas were than calculated and if they were larger than 10, they were included as a numbered rectangle. Other rectangles were excluded, coloured black here.*

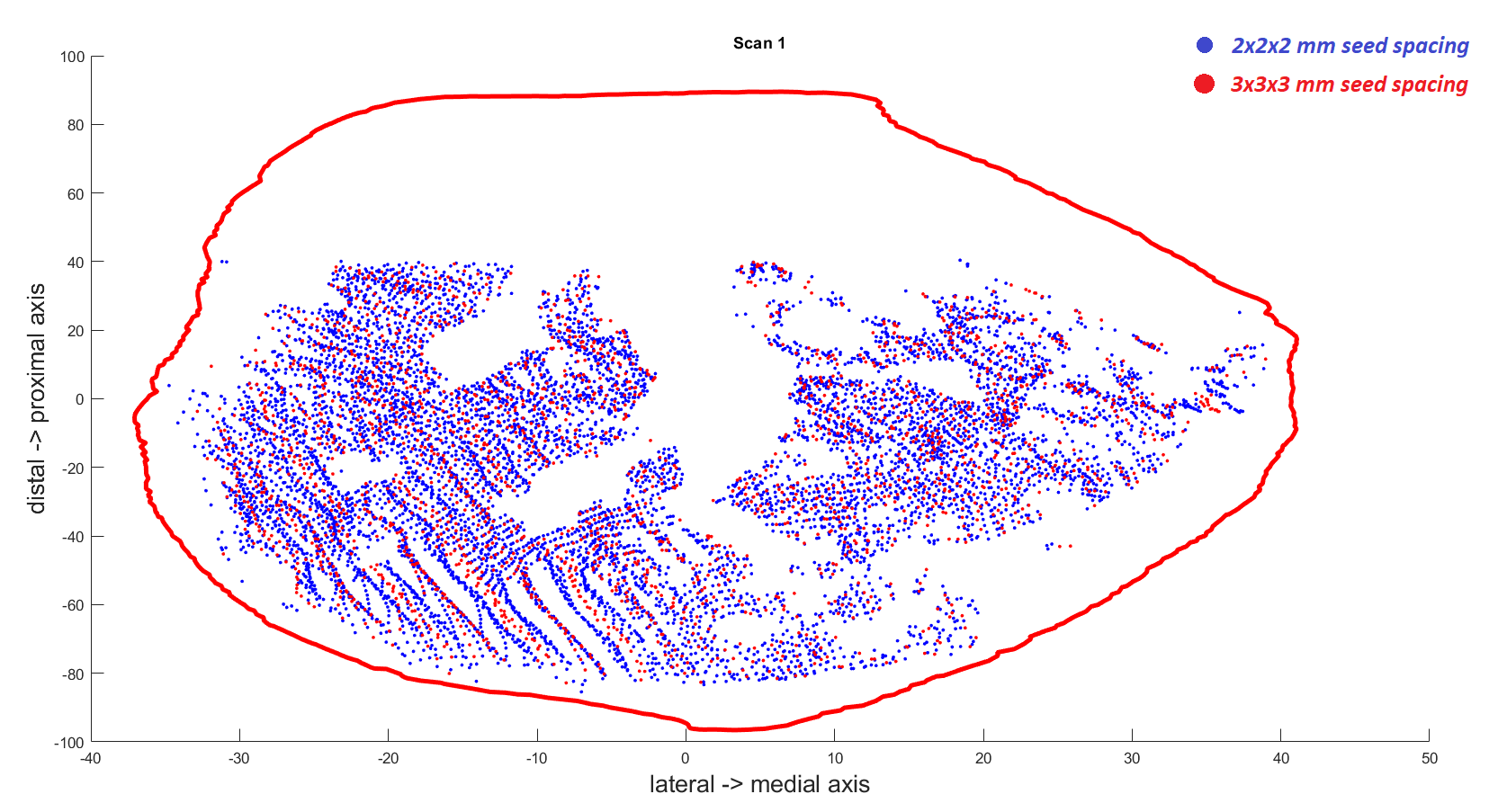
For each rectangle size, the figure shown in G9 was shown for both scans with the numbers shown in the rectangles. These numbers were then visually compared between the two scans and rectangles that did not match were deleted. These were almost always in the “corners” of the muscle, where sometimes the segmentation in one scan had slightly more surface area in a rectangle, passing the threshold of 10 and thus including the rectangle. In the other scan, it would then fall just below the threshold and not be included. This then shifts the numbers assigned to the rectangles by 1 (Fig. G10 shows an example). This was a major issue because these numbers were needed in order to calculate correlations between the rectangles for scan 1 and 2 that was used to determine the reliability of the measures. Therefore, care was taken in this step to make sure no mistakes were made here and a double inspection of all rectangles for all sizes and participants was performed before continuing with the data for correlations and further analyses.



*Figure G10: Example of an incorrect automatic labelling of the rectangles. Rectangle #38 (you may need to zoom in 😊) was assigned to the second rectangle from the left in scan 1 (left figure), whereas it was assigned to the first rectangle from the left in scan 2 (right figure). This is because the slight difference in segmentation. In scan 2, the surface area of the “within muscle area” for this rectangle was larger than the set threshold, while is was just below the threshold in the first rectangle from the left in scan 1, therefore it was not counted as a rectangle in scan 1 (left figure). This causes the number for each rectangle after this one to be off by 1 between both scans, which will have a very large effect on the correlations. Therefore, rectangle #38 would be manually deleted here for scan 2, so that here #38 is also the second rectangle from the left. Note that the same needed to be done here for #46.*

**Seed spacing**

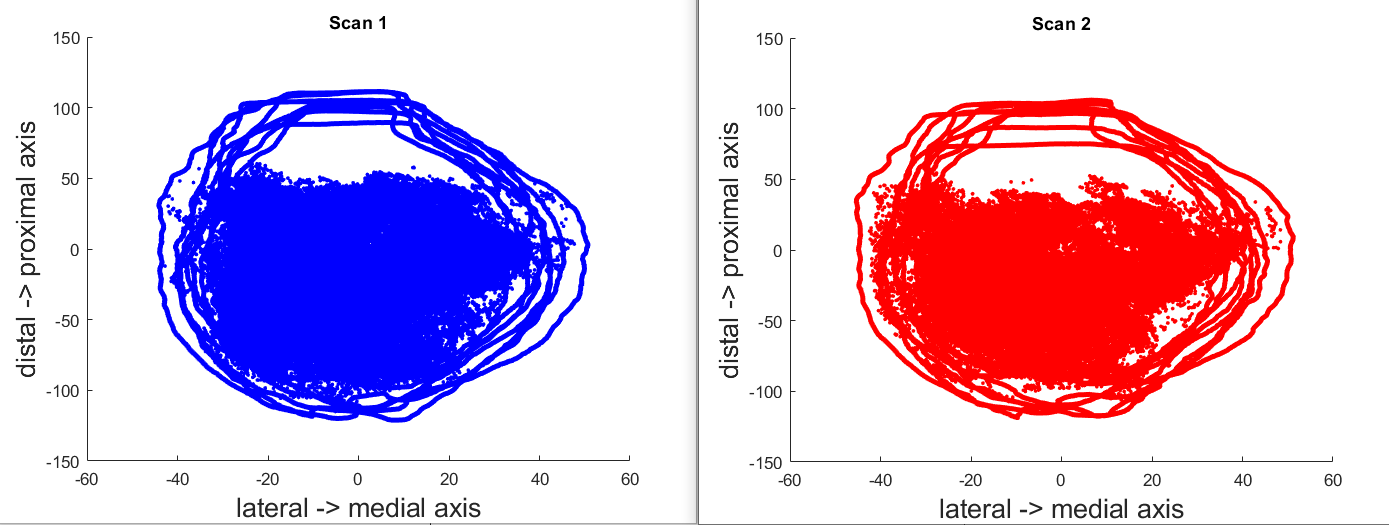
We compared different seed spacing, namely 2x2x2 mm with the previously used 3x3x3 mm. Pearson correlation coefficients were compared and showed no (visually) significant improvements consistently across the different rectangle sizes. The location of the midpoints between the 2 and 3 mm seed spacing results were also compared visually. No important difference was found here, more specifically no previously “empty” areas were filled when using a higher resolution seed spacing. The figure below (Fig S1) shows an example of a typical subject. We continued with the 2x2x2 mm spacing.



*Figure S1: comparison between fibre midpoint locations between 2mm (blue dots) and 3mm (red dots) seed spacing. Note that the vertical axis label is flipped here…*

**Fibre tract locations**

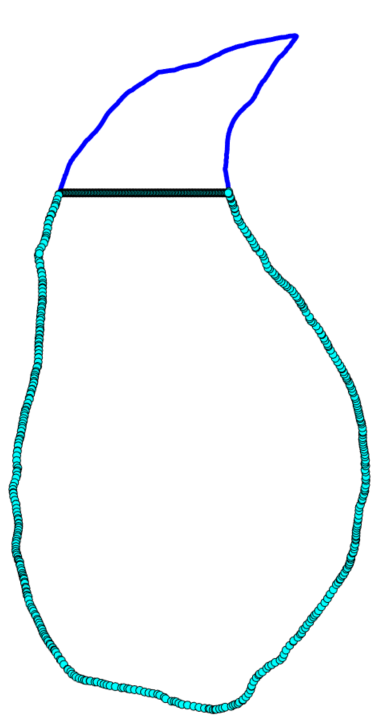
The figures below (Fig. S2) show the results from a quick test to see if there was a systematic factor between participants in the regions of the muscle for which no midpoints were found. If this were true, then we would expect to see “white gaps” such as shown in figure S1 above for the middle section of the muscle. It appears this is not the case, confirming that these “gaps” are likely due to individual anatomical variations.



*Figure S2: Overlayed muscle boundaries and midpoints of fibres for all 8 subjects and both scans. If there was a systematic factor to the regions of the muscle where no fibre tracts were detected, we would expect to see empty white gaps in these figures.*

**Proximal segmentation cut**

The segmentations of the reliability data were terminated earlier (more distally) compared to the current data because of the different field of view in the two scanning procedures. This caused an issue to compare each region of interest (grid method) between the reliability and the current data. We therefore cut the current data segmentations proximally as well. For this we first assessed the approximate location of where the reliability data were cut off. This was assessed visually using both scans from all 8 participants of the reliability data.



*Figure Y3: Example of cut-off for segmentation to better match reliability segmentations.*

1. https://doi.org/10.1016/j.jbiomech.2019.01.043 [↑](#footnote-ref-1)