Supplementary Information: ImageJ Macro for Monocyte Localization

```
/*
     Protocol:
     User traces a region of interest corresponding to lymphatics and adds to ROI Manager by
pressing "t" or by Edit > Selection > Add to Manager.
     Add one or more other regions (T zone).
     macro "Save ROI Manager [F7]"
     macro "Measure traced areas [F6]"

*/

var threshold_object_minsize = 3;              // exclude objects too small to be considered cells
var threshold_channel = "";                    //  sets test type for output string
var measure_channel = "";                      //  sets text type for output string
var thresholds = newArray(32, 40, 50);         //  lower bound for the thresholds for tallying cells
var tallies = newArray (thresholds.length);    //  array to hold the counts (results) at each
threshold
var tab = "\t ";
var print_header = true;                       //  first time run, label the results columns


macro "Measure traced areas [F6]"  {
  threshold_channel = "C2-";             // segmentation on this channel
  measure_channel = "C4-";               // measurement on this channel
  requires("1.51t");
  original = getImageID;
  tracedAreas = roiManager("count");
  if (tracedAreas < 2)                   // must be lymph node and at least one sub-area
    exit("ROI Manager must contain traced areas.");
  roiManager("Save", "tempROIs.zip");
  title = getTitle;
  run("Split Channels");
  threshold_channel  = "" + threshold_channel  + title;
  measure_channel  = "" + measure_channel  + title;
  if (print_header) {                    // first time run, label the results columns
    print_header = false;
    output = "thesholds_at " + tab +" "+tab+"total#" ;
    for (i=0; i<tallies.length; i++)
      output=output+tab+thresholds[i];
    print(output);
  }
  selectWindow(threshold_channel);
  run("Select None");
  run("Median...", "radius=1");
  run("Threshold...");
  setAutoThreshold("Default dark");
  waitForUser("Adjust Threshold", "Adjust threshold and click OK in this window.  \nYou may
zoom, pan, etc.  \nDo not click Apply in Threshold window\nClick ok must be in this window.");
  getThreshold(lower, upper);
  run("Set Measurements...", "area mean display redirect=None decimal=2");
```

```
  // MEASURE LYMPHATICS
  selectWindow(threshold_channel);
  roiManager("Select", 0);                // First ROI is always considered the Lymphatics region.
  report_in_big_area("lymphatics");

    // MEASURE FIRST T ZONE
    roiManager("reset");
    roiManager("Open", "tempROIs.zip");
    selectWindow(threshold_channel);
    roiManager("Select", 1);
    report_in_big_area("t-zone");


    // If there is one, MEASURE SECOND T ZONE
    roiManager("reset");
    roiManager("Open", "tempROIs.zip");
    if (roiManager("count") == 3) {
      selectWindow(threshold_channel);
      roiManager("Select", 2);
      report_in_big_area("t-zone");
    }
  close("Results");                       // clean up open windows
  close("C*");
  roiManager("reset");
  selectWindow("Log");                    // show the results
}  // end macro Measure traced areas


/* ================================================
Reports number of cells based on intensities threshold in active ROI.
*/
function report_in_big_area(output_label) {
    setThreshold(lower, upper);
    run("Analyze Particles...", "size="+threshold_object_minsize+"-Infinity exclude clear include
add");
    selectWindow(measure_channel);
    Array.fill(tallies, 0);
    output = output_label + tab + roiManager("count");
    for (i=0; i<roiManager("count"); i++) {
      roiManager("select", i);
      run("Measure");
      tally();
    }
    for (j=0; j<tallies.length; j++) {
      output = output + tab + tallies[j];
    }
    print (title + tab + output);
}  // end function report_in_big_area()
```

```
/* ================================================
This function increments the counts of objects at the different intensity thresholds.
*/
function tally() {
  intensity = getResult("Mean", nResults-1);
  for (j=0; j<tallies.length; j++) {
    if (intensity > thresholds[j])
      tallies[j]++;
  }
}  // end function tally()


/*  This macro saves the ROI Manager data with the same file name as the image.  It only works
when the image was opened with ImageJ, not BioFormats, such that getDirectory("image") will
be set correctly.
*/
macro "Save ROI Manager [F7]" {
  if ( getInfo("window.type") != "Image" )
    exit("Make sure running macro from image window.");
  dir = getDirectory("image");                                    // where to save results
  if (lengthOf(dir) < 1) exit("Make sure the image was opened from disk.");
  title = getTitleStripExtension();
  newName =dir + title + ".zip";
  roiManager("Save", newName);
  roiManager("reset");                                            // optional depending on work flow
}  // end Save ROI Manager [F7]



/* Use this function to strip any number of extensions off images.
Returns the title without the extension.
*/
function getTitleStripExtension() {
  t = getTitle();
  extensions = newArray(".tif", ".tiff", ".lif", ".lsm", ".czi", ".nd2", ".ND2");
  for(i=0; i<extensions.length; i++)
    t = replace(t, extensions[i], "");
  return t;
}  // end getTitleStripExtension()
```