Electronic Supplementary Information

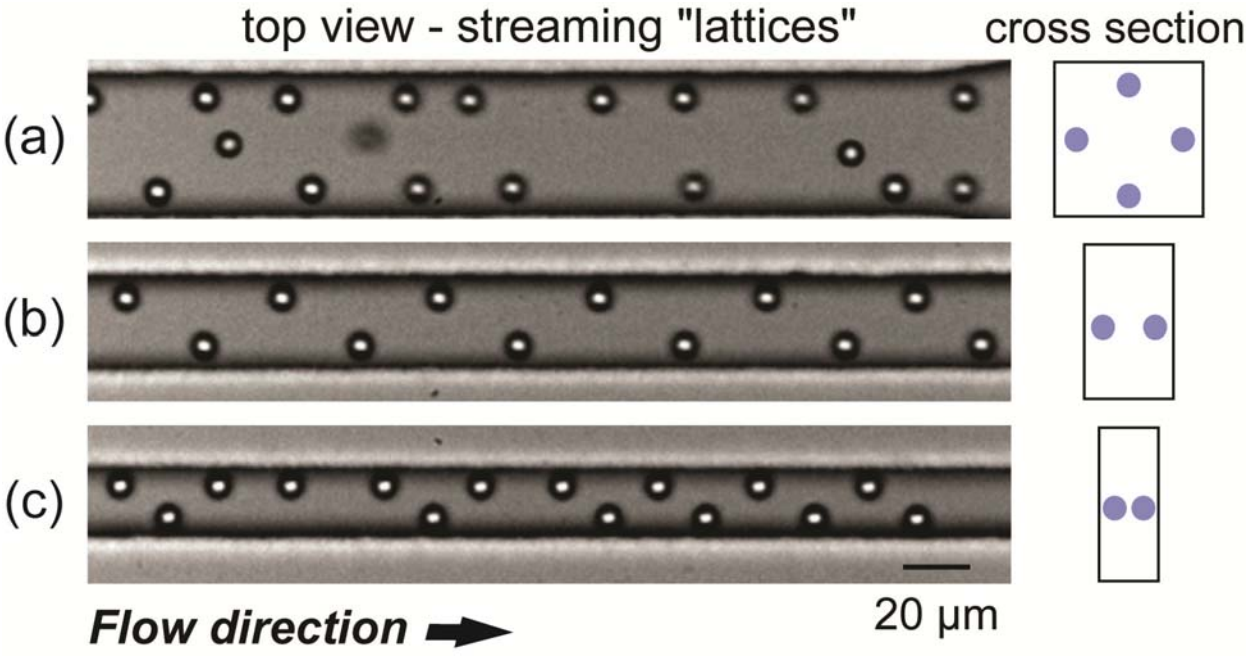
Sheathless Inertial Cell Ordering For Extreme Throughput Flow Cytometry

Soojung C. Hur[1], Henry T.K. Tse[2] and Dino Di Carlo[2,3]

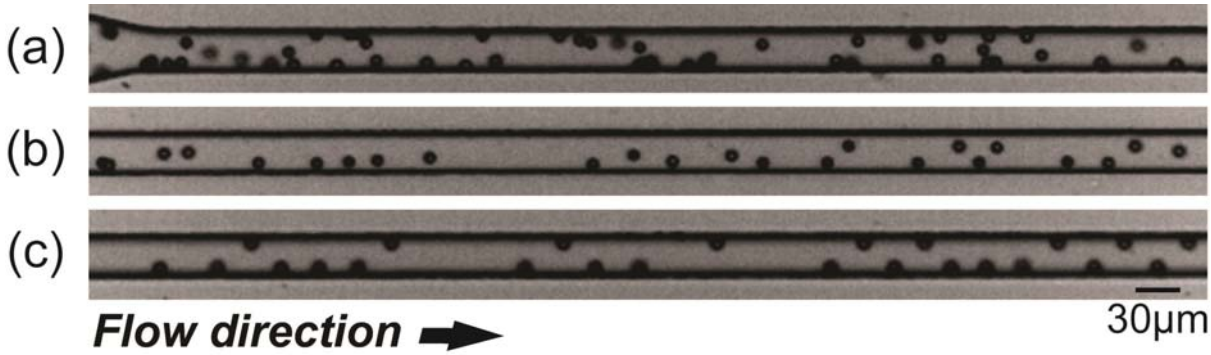
1 Mechanical and Aerospace Engineering Department, University of California Los Angeles, Los Angele, CA, USA 90095
2 Department of Bioengineering, University of California Los Angeles, Los Angele, CA, USA 90095
3 California NanoSystems Institute, Los Angeles, CA, USA 90095

Table of Contents

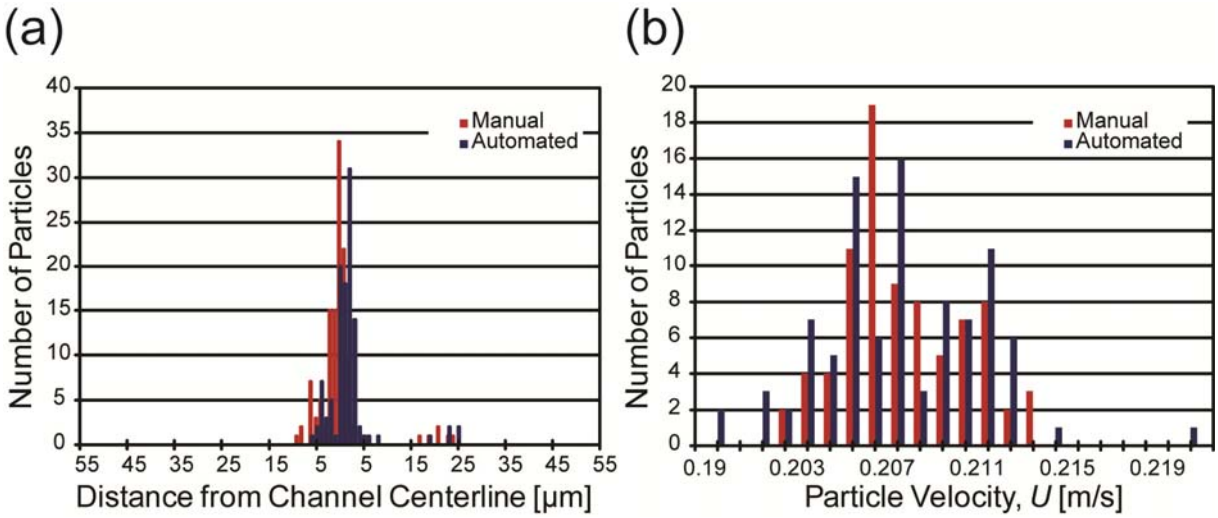
P.
ESI Figure 1…… S2
ESI Figure 2…… S3
ESI Figure 3…… S4
ESI Movie Captions…….. S5
ESI MATLAB Script…… S6



ESI_Figure 1 Aspect ratio effects on inertial ordering and focusing. High speed microscopy images of 10 µm polystyrene particles ordered in (a) 50×50 µm, (b) 30×50 µm, and (c) 20×50 µm channels. Particles are flowing from left to right at U ~10 cm/sec. Images are taken 4 cm downstream of the particle inlet. Ordering and focusing accuracy is seen to increase with decreasing channel width. Inter-particle distance in the stable ordering lattice is seen to decrease with decreasing channel dimensions.



ESI_Figure 2 Development of inertial ordering in a microchannel. Three high speed images, taken at (a) the inlet, (b) 5mm downstream and (c) 1cm downstream, are shown of 10 μm particles flowing in a 30×50 μm channel at U ~10 cm/sec. At the inlet the particles are randomly distributed. After traveling 5 mm downstream particles have become more ordered and few overlapping particles are observed. After traveling 1 cm downstream in the straight channel, equilibrium ordering positions are mostly achieved.



ESI_Figure 3 Comparison between manual and automated image analysis results for (a) distance between the particle center and the channel wall and (b) particle velocity.

ELECTRONIC SUPPLIMENTARY INFORMATION MOVIES CAPTIONS

ESI_Movie 1 Massively parallel cell positioning microfluidic device. 5%v/v dilute blood flowing at Q = 2.5ml/min.

ESI_Movie 2 Comparison between unfocused and focused cells at Q = 50µl/min and 2.5ml/min, respectively.

ESI_Movie 3 Change in z-positions as function of particle Reynolds number, R_p.

ELECTRONIC SUPPLIMENTARY INFORMATION MATLAB SCRIPT

```
% RBC & WBC particle positioning
% 2009_09_28
% To use the script, kernel images matching to be identify
% particles must be provided
% Henry Tse
% henrytse@ucla.edu

close all
clear all
Result_RBC=[];
Result_WBC=[];
tt=1;
zz=1;
mm=1;
IN=input('Total number of images to analyze:');
for tt=1:IN
    clc
%    Skip frame counter
    zz=5*tt-4;
    fileIndex=zz;
    CountDown=IN-tt;
    display(CountDown);
for mm=1:4 %quadrant loop
clear i I New M itop ibot i_new x1 x2 a_max b_max a b pk pk5 pk6 W_W W_R;
bead_locations=[];
% IN=1;
% Load kernel images for RBCs and WBCs
kernel{1}=imread('rbc_kernel_1.tif');
kernel{2}=imread('wbc_kernel_1.tif');
k3=imread('rbc_kernel_2.tif');
k4=imread('rbc_kernel_3.tif');
k5=imread('rbc_kernel_4.tif');
k2=kernel{2};
k6=imread('wbc_kernel_5.tif');
k6_2=imread('wbc_kernel_4.tif');
if size(k2,3)>1
    k2(:,:,2)=[];
    k2(:,:,2)=[];
end
if size(k3,3)>1
    k3(:,:,2)=[];
    k3(:,:,2)=[];
end
if size(k4,3)>1
    k4(:,:,2)=[];
    k4(:,:,2)=[];
end
if size(k5,3)>1
    k5(:,:,2)=[];
    k5(:,:,2)=[];
end
if size(k6,3)>1
    k6(:,:,2)=[];
    k6(:,:,2)=[];
end
if size(k6_2,3)>1
    k6_2(:,:,2)=[];
```

```
    k6_2(:,:,2)=[];
end
FF=[0.40, 0.55, 0.80, 0.4]; % fitting factor corresponding to kernel(1), kernel(2)-upper, kernel(2)-lower, kernel(2)-%
diff
i=[];
% i=imread('WBCgal.tif');
%batch image processing
p=which('2500uL_light_page_0001.tif');
filelist=dir([fileparts(p) filesep '2500uL_light_page_*.tif']);
fileNames = {filelist.name}';

xx=[0,0,370,370];
yy=[0,301,0,301];

i=imread(fileNames{fileIndex});
i=imcrop(i,[0 0 800 600]);

if mm==1
    i=imcrop(i,[xx(mm) yy(mm) xx(mm)+400 yy(mm)+300]);
end
if mm==2
    i=imcrop(i,[xx(mm) yy(mm) xx(mm)+400 yy(mm)+300]);
end
if mm==3
    i=imcrop(i,[xx(mm) yy(mm) xx(mm)+429 yy(mm)+330]);
end
if mm==4
    i=imcrop(i,[xx(mm) yy(mm) xx(mm)+429 yy(mm)+300]);
end
W1=[];
ik=1;
% convert to double for corr/conv processing
k=kernel{ik};
if size(k,3)>1
    k(:,:,2)=[];
    k(:,:,2)=[];
end
I=double(i);
M=max(max(I));
New=I/M;
New_2=New;
i=double(i);
i=i/M;
K=[];
% Image Filters
k5=imadjust(im2double(k5),[],[0.01 1]);
k6=imadjust(im2double(k6),[],[0.01 1]);
k6_2=imadjust(im2double(k6_2),[],[0.01 1]);
se=strel('disk',5);
itop=imtophat(i,se);
ibot=imbothat(i,se);
i_new=imsubtract(imadd(itop,i),ibot);
ibot=double(ibot);
ibot=ibot/max(max(ibot));
for a=1:size(ibot,1);
    for b=1:size(ibot,2);
        if ibot(a,b)==0;
            ibot(a,b)=0.01;
        end
    end
end
```

```
K2=[];
K2=k2;
K2=double(K2);
J2=max(max(K2));
k2=K2/J2;
k3=double(k3);
k3=k3/max(max(k3));
k4=double(k4);
k4=k4/max(max(k4));
K=k;
K=double(K);
J=max(max(K));
k=K/J;
loc=[0,0];
% Collecting Sizes
x1=zeros(size(K,1),size(K,2));
x1(round(size(k,1)/2),round(size(k,2)/2))=1;
x2=zeros(size(K2,1),size(K2,2));
x2(round(size(k2,1)/2),round(size(k2,2)/2))=1;

c=2;
a_max=size(New,1)-round(1.5*size(k,1)-1);
b_max=size(New,2)-round(1.5*size(k,2)-1);
a2_max=size(New,1)-round(1.5*size(k2,1))-1;
b2_max=size(New,2)-round(1.5*size(k2,2))-1;
a_max=min(a_max,a2_max);
b_max=min(b_max,b2_max);

a=round(1.5*size(k,1))+1;
b=round(1.5*size(k,2))+1;
b_skip=[round((size(K,1)-1)/2)+5];
j=[0:size(k2,1)];
sum_k=sum(sum(k));
i3=[];
pk=size(k,1)*size(k,2);
pk2=size(k2,1)*size(k2,2);
pk3=size(k3,1)*size(k3,2);
pk4=size(k4,1)*size(k4,2);
pk5=size(k5,1)*size(k5,2);
pk6=size(k6,1)*size(k6,2);
W_R=zeros(size(New,1),size(New,2));
W_W=zeros(size(New,1),size(New,2));
W2=zeros(size(New,1),size(New,2));
i1=1;i2=1;i4=1;i5=1; i13=1; i23=1; i14=1; i24=1;

%RBC ID%
FF6=0.85; %max rbx value
FF26=0.28; %min rbc value%
FF20=0.015; %Difference from middle to wall column
FF21=0.665; %max intensity allowed
FF22=0.1; %difference % between image and kernel
FF23=0.25; %min rbc value
FF24=0.15; %rogue RBC kernel 3
FF10 = 0.1;

while a<=a_max
   if (sum(New(a+round(size(k,1)/2)-1,1:size(New,2)))>250)
         while b<=b_max
         for i1=1:size(k5,1)
            for i2=1:size(k5,2)
```

```
            i3(i1,i2)=(abs(ibot(a+i1-1,b+i2-1)-k5(i1,i2))/(ibot(a+i1-1,b+i2-1)+k5(i1,i2)));
        end
    end
    for i1=1:size(k,1)
        for i2=1:size(k,2)
            i3(i1,i2)=(abs(New(a+i1-1,b+i2-1)-k(i1,i2))/k(i1,i2));
        end
    end
    cc=0;
    if (sum(sum(i3))<0.35*pk5 & sum(sum(i3))<0.35*pk ...
            & ibot(a+round(size(k,1)/2),b+2) <=FF6 & ibot(a+round(size(k,1)/2),b+size(k,2)-2)<=FF6 ...
            & ibot(a+1,b+round(size(k,2)/2))<=FF6 & ibot(a+size(k,1)-1,b+round(size(k,2)/2))<=FF6 ...
            & (sum(sum(W_R(a-round(.5*size(k,1)):a+round(1.5*size(k,1)),b-
round(.5*size(k,2)):b+round(1.5*size(k,2)))))<1 ...
            & ibot(a-1,b-1)<FF26 & ibot(a+size(k,1)+1,b+size(k,2)+1)<FF26 ...
            & abs(sum(New(a+round(size(k,1)/2),b:b+size(k,2))-New(a+round(size(k,1)/2),b-
size(k,2):b))/sum(New(a+round(size(k,1)/2),b:b+size(k,2))+New(a+round(size(k,1)/2),b-size(k,2):b))) >
0.075)) %intensity diference of middle to previous middle
        if (ibot(a+round(size(k,1)/2),b+2) >=FF26 |ibot(a+round(size(k,1)/2),b+3) >=FF26 ...
            |ibot(a+round(size(k,1)/2),b+1) >=FF26 )
            cc=cc+1;
        end
        if (ibot(a+round(size(k,1)/2),b+size(k,2)-2)>=FF26 |ibot(a+round(size(k,1)/2),b+size(k,2)-3)>=FF26 ...
                |ibot(a+round(size(k,1)/2),b+size(k,2)-1)>=FF26 )
            cc=cc+1;
        end
        if (ibot(a+1,b+round(size(k,2)/2))>=FF26 | ibot(a,b+round(size(k,2)/2))>=FF26 ...
                | ibot(a+2,b+round(size(k,2)/2))>=FF26)
            cc=cc+1;
        end
        if (ibot(a+size(k,1),b+round(size(k,2)/2))>=FF26 | ibot(a+size(k,1)-1,b+round(size(k,2)/2))>=FF26 ...
            | ibot(a+size(k,1)-2,b+round(size(k,2)/2))>=FF26)
            cc=cc+1;
        end
        if abs(sum(New(a:a+size(k,1),b)-
New(a:a+size(k,1),b+round(size(k,2)/2)))/sum((New(a:a+size(k,1),b)+New(a:a+size(k,1),b+round(size(k,2)/2)))) >
FF10
            cc=cc+1;
        end
        if cc>3
            W_R(a:a+size(K,1)-1,b:b+size(K,2)-1)=x1;
            W2(a:a+size(K,1)-1,b:b+size(K,2)-1)=1;
        end
    end
    b=b+1;
  end
  end
  a=a+1;
  b=round(1.5*size(k,2))+1;
  ii=1;
end

for a=1:size(New,1)
  if sum(New(a,1:size(New,2)))<170
    New(a,1:size(New,2))=1;
  end
end

%WBC ID EOSINOPHIL COUNT%
FF7=0.80; %percent of wbc kernel
%FF(2) upper bound, FF(3) lower bound of wbc kernel
```

```
FF8=0.89; %upper bound at walls
FF9=0.45; %lower bound at walls
FF10=0.02; %differences between left wall to middle column
FF11=0.02; %difference between top wall and middle row
FF12=0.025; %differnece between current middle of image to previous middle
FF15=100; %im2bw(image,0.75) must be greater than, to cancel out unfocused cells
FF16=180; %im2bw(image,0.75) must be loewr than, "
FF17=50; %im2bw(across a-2:a+2 at middle) must be greater than
FF18=25; %Lower bound for FF17
FF19=135; %Half image im2bw min
FF25=150; %threshold of image at 0.7 min sum
FF27=0.20; %ibot wbc threshold
FF28=0.045; % percent change of middle _+2
FF29=2.2; % imcrop compare mean
FF30=1000; % imcrop compare sum
      cc=0;
a=round(1.5*size(k2,1))+1;
b=round(1.5*size(k2,2))+1;
k6_1=k6;
while a<=a_max
   while b<=b_max
      k6=k6_1;
      for i4=1:size(k6,1)
         for i5=1:size(k6,2)
            i6(i4,i5)=(abs(ibot(a+i4-1,b+i5-1)-k6(i4,i5))/(ibot(a+i4-1,b+i5-1)+k6(i4,i5)));
         end
      end
      cc=0;
      if (sum(sum(i6))<0.4*pk6 & sum(sum(New(a:a+size(k2,1)-1,b:b+size(k2,2)-1)))/(pk2) > FF(2) ...
            & sum(sum(New(a:a+size(k2,1)-1,b:b+size(k2,2)-1)))/(pk2) < FF(3) ...
            & sum(sum(New(a:a+size(k2,1)-1,b:b+size(k2,2)-1))) > FF7*sum(sum(k2))...
            & New(a+round(size(k2,1)/2),b+2) < FF8 & New(a+round(size(k2,1)/2),b+size(k2,2)-2)< FF8 ...
            & New(a+round(size(k2,1)/2),b+2) > FF9-.1 & New(a+round(size(k2,1)/2),b+size(k2,2)-2)> FF9-.1 ...
            & New(a+2,b+round(size(k2,2)/2))> FF9  ...
            & ( New(a+size(k2,1)-2,b+round(size(k2,2)/2))> FF9| New(a+size(k2,1)-3,b+round(size(k2,2)/2))> FF9)...
            & New(a+2,b+round(size(k2,2)/2))< FF8  ...
            & (max(max(ibot(a+round(size(k2,1)/2)-3:a+round(size(k2,1)/2)+3,b+round(size(k2,2)/2)-
3:b+round(size(k2,2)/2+3)))))<0.6 ... % max intensity at 9x9 middle
            &( New(a+size(k2,1)-2,b+round(size(k2,2)/2))< FF8 | New(a+size(k2,1)-3,b+round(size(k2,2)/2))< FF8 )...
            & abs(sum(New(a:a+size(k2,1),b)-
New(a:a+size(k2,1),b+round(size(k2,2)/2)))/sum((New(a:a+size(k2,1),b)+New(a:a+size(k2,1),b+round(size(k2,2)/2)))))
> FF10 ...
            & abs(sum(ibot(a,b:b+size(k2,2))-
ibot(a+round(size(k2,1)/2),b:b+size(k2,2)))/sum(ibot(a,b:b+size(k2,2))+ibot(a+round(size(k2,1)/2),b:b+size(k2,2)))) >
FF11 ...
            & abs(sum(New(a+round(size(k2,1)/2),b:b+size(k2,2))-New(a+round(size(k2,1)/2),b-
size(k2,2):b))/sum(New(a+round(size(k2,1)/2),b:b+size(k2,2))+New(a+round(size(k2,1)/2),b-size(k2,2):b))) > FF12 ...
            & sum(sum(W_W(a-round(1*size(k2,1)):a,b-round(1*size(k2,2)):b))) < 1 ...
            & (max(New(a+round(size(k2,1)/2),b:b+round(size(k2,2))))-
min(New(a+round(size(k2,1)/2),b:b+round(size(k2,2)))))...

/(max(New(a+round(size(k2,1)/2),b:b+round(size(k2,2))))+min(New(a+round(size(k2,1)/2),b:b+round(size(k2,2)))))>0.
15 ... % intensity diference
            & (max(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2)))-
min(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2))))...

/(max(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2)))+min(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2)))
)<0.35 )...
            if (ibot(a+round(size(k2,1)/2),b+1) >=FF27 |ibot(a+round(size(k2,1)/2),b+2) >=FF27
|ibot(a+round(size(k2,1)/2),b+3) >=FF27 ) ...
               cc=cc+1;
```

```
            end
            if (ibot(a+round(size(k2,1)/2),b+size(k2,2)-1)>=FF27 |ibot(a+round(size(k2,1)/2),b+size(k2,2)-2)>=FF27
|ibot(a+round(size(k2,1)/2),b+size(k2,2)-3)>=FF27 ) ...
                cc=cc+1;
            end
            if (ibot(a+1,b+round(size(k2,2)/2))>=FF27 | ibot(a,b+round(size(k2,2)/2))>=FF27 ...
                | ibot(a+2,b+round(size(k2,2)/2))>=FF27)
               cc=cc+1;
            end
            if (ibot(a+size(k2,1),b+round(size(k2,2)/2))>=FF27 | ibot(a+size(k2,1)-1,b+round(size(k2,2)/2))>=FF27 ...
                | ibot(a+size(k2,1)-2,b+round(size(k2,2)/2))>=FF27)
               cc=cc+1;
            end
            if cc>=3
               W_W(a:a+size(K2,1)-1,b:b+size(K2,2)-1)=x2;
               W1(a:a+size(K2,1)-1,b:b+size(K2,2)-1)=1;
            end
        end
        k6=k6_2;
        for i4=1:size(k6,1)
           for i5=1:size(k6,2)
               i6(i4,i5)=(abs(ibot(a+i4-1,b+i5-1)-k6(i4,i5))/(ibot(a+i4-1,b+i5-1)+k6(i4,i5)));
           end
        end
        cc=0;
        if (sum(sum(i6))<0.55*pk6 & sum(sum(New(a:a+size(k2,1)-1,b:b+size(k2,2)-1)))/(pk2) > FF(2) ...
            & sum(sum(New(a:a+size(k2,1)-1,b:b+size(k2,2)-1)))/(pk2) < FF(3) ...
            & sum(sum(New(a:a+size(k2,1)-1,b:b+size(k2,2)-1))) > FF7*sum(sum(k2))...
            & New(a+round(size(k2,1)/2),b+2) < FF8 & New(a+round(size(k2,1)/2),b+size(k2,2)-2)< FF8 ...
            & New(a+round(size(k2,1)/2),b+2) > FF9-.1 & New(a+round(size(k2,1)/2),b+size(k2,2)-2)> FF9-.1 ...
            & New(a+2,b+round(size(k2,2)/2))> FF9  ...
            & ( New(a+size(k2,1)-2,b+round(size(k2,2)/2))> FF9| New(a+size(k2,1)-3,b+round(size(k2,2)/2))> FF9)...
            & New(a+2,b+round(size(k2,2)/2))< FF8  ...
            & max(max(New(a:a+size(k,1)-1, b:b+size(k,2)-1)))<0.98 ...
            & max(max(ibot(a+round(size(k2,1)/2),b+round(size(k2,2)/2)-3:b+round(size(k2,2)/2)+3))) < 0.1  ... %
intensity of extra bright RBC
            &( New(a+size(k2,1)-2,b+round(size(k2,2)/2))< FF8 | New(a+size(k2,1)-3,b+round(size(k2,2)/2))< FF8 )...
            & abs(sum(New(a:a+size(k2,1),b)-
New(a:a+size(k2,1),b+round(size(k2,2)/2)))/sum((New(a:a+size(k2,1),b)+New(a:a+size(k2,1),b+round(size(k2,2)/2)))))
> FF10 ...
            & abs(sum(ibot(a,b:b+size(k2,2))-
ibot(a+round(size(k2,1)/2),b:b+size(k2,2)))/sum(ibot(a,b:b+size(k2,2))+ibot(a+round(size(k2,1)/2),b:b+size(k2,2)))) >
FF11 ...
            & abs(sum(New(a+round(size(k2,1)/2),b:b+size(k2,2))-New(a+round(size(k2,1)/2),b-
size(k2,2):b))/sum(New(a+round(size(k2,1)/2),b:b+size(k2,2))+New(a+round(size(k2,1)/2),b-size(k2,2):b))) > FF12 ...
            & sum(sum(W_W(a-round(1*size(k2,1)):a,b-round(1*size(k2,2)):b))) < 1 ...
            & (sum(ibot(a+round(size(k2,1)/2)-1,b+round(size(k2,2)/2)-4:b+round(size(k2,2)/2)+4))< 0.40 ...
            |sum(ibot(a+round(size(k2,1)/2),b+round(size(k2,2)/2)-4:b+round(size(k2,2)/2)+4))< 0.40  ) ...% intensity at
middle row
            & sum(ibot(a+round(size(k2,1)/2)-4:a+round(size(k2,1)/2)+4,b+round(size(k2,2)/2))) < 0.40 ...
            & (max(max(ibot(a+round(size(k2,1)/2)-3:a+round(size(k2,1)/2)+3,b+round(size(k2,2)/2)-
3:b+round(size(k2,2)/2+3)))))<0.2 ... % max 9x9 in middle
            & sum(sum(imdivide(imcrop(ibot,[b a size(k6,2)-1 size(k6,1)-1]),k6)))<1000 &
mean(mean(imdivide(imcrop(ibot,[b a size(k6,2)-1 size(k6,1)-1]),k6)))< FF29 ...
            & (max(New(a+round(size(k2,1)/2),b:b+round(size(k2,2))))-
min(New(a+round(size(k2,1)/2),b:b+round(size(k2,2)))))...

/(max(New(a+round(size(k2,1)/2),b:b+round(size(k2,2))))+min(New(a+round(size(k2,1)/2),b:b+round(size(k2,2)))))>0.
15 ... % intensity diference
            & (max(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2)))-
min(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2))))...
```

```
/(max(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2)))+min(New(a:a+round(size(k2,1)/2),b+round(size(k2,2)/2)))
)<0.35 )...
            if (ibot(a+round(size(k2,1)/2),b+1) >=FF27 |ibot(a+round(size(k2,1)/2),b+2) >=FF27
|ibot(a+round(size(k2,1)/2),b+3) >=FF27 ) ...
                cc=cc+1;
            end
            if sum(abs(New(a+1,b+5:b+size(k2,1)-6)-New(a+round(size(k2,1)/2),b+5:b+size(k2,1)-
6)))/sum((abs(New(a+1,b+5:b+size(k2,1)-6)+New(a+round(size(k2,1)/2),b+5:b+size(k2,1)-6)))) > 0.03 % intensity at
middle
                cc=cc+1;
            end
            if (ibot(a+round(size(k2,1)/2),b+size(k2,2)-1)>=FF27 |ibot(a+round(size(k2,1)/2),b+size(k2,2)-2)>=FF27
|ibot(a+round(size(k2,1)/2),b+size(k2,2)-3)>=FF27 ) ...
                    cc=cc+1;
            end
            if (ibot(a+1,b+round(size(k2,2)/2))>=FF27 | ibot(a,b+round(size(k2,2)/2))>=FF27 ...
                    | ibot(a+2,b+round(size(k2,2)/2))>=FF27)
                cc=cc+1;
            end
            if (ibot(a+size(k2,1),b+round(size(k2,2)/2))>=FF27 | ibot(a+size(k2,1)-1,b+round(size(k2,2)/2))>=FF27 ...
                    | ibot(a+size(k2,1)-2,b+round(size(k2,2)/2))>=FF27)
                cc=cc+1;
            end

            if cc>=4
                W_W(a:a+size(K2,1)-1,b:b+size(K2,2)-1)=x2;
                W1(a:a+size(K2,1)-1,b:b+size(K2,2)-1)=1;
            end
        end
        b=b+1;
    end
    a=a+1;
    b=round(1.5*size(k2,2))+1;
    ii=1;
end
j1=1;
j2=1;
ii=1;

W_R(1:size(k(1)),1:size(k(2)))=zeros(1:size(k(1)),1:size(k(2)));
New=New_2;
a=round(1.5*size(k,1))+1;
b=round(1.5*size(k,2))+1;

%Filtering RBC and RBC counts
while a<=a_max
    if (sum(New(a+round(size(k,1)/2)-1,1:size(New,2)))>250)
        while b<=b_max
        for i1=1:size(k5,1)
            for i2=1:size(k5,2)
                i3(i1,i2)=(abs(ibot(a+i1-1,b+i2-1)-k5(i1,i2))/(ibot(a+i1-1,b+i2-1)+k5(i1,i2)));
            end
        end
        for i1=1:size(k,1)
            for i2=1:size(k,2)
                i3(i1,i2)=(abs(New(a+i1-1,b+i2-1)-k(i1,i2))/k(i1,i2));
            end
        end
        cc=0;
        if (sum(sum(i3))<0.55*pk5 & sum(sum(i3))<0.5*pk ...
```

```
                & ibot(a+round(size(k,1)/2),b+2) <=FF6 & ibot(a+round(size(k,1)/2),b+size(k,2)-2)<=FF6 ...
                & ibot(a+1,b+round(size(k,2)/2))<=FF6 & ibot(a+size(k,1)-1,b+round(size(k,2)/2))<=FF6 ...
                & sum(sum(W_W(a-round(1*size(k,1)):a,b-round(1.5*size(k,2)):b))) < 1 ...
                & abs(sum(New(a+round(size(k,1)/2),b:b+size(k,2))-New(a+round(size(k,1)/2),b-
size(k,2):b))/sum(New(a+round(size(k,1)/2),b:b+size(k,2))+New(a+round(size(k,1)/2),b-size(k,2):b))) > 0.045 ... %
comparing middle intensities
                & (sum(sum(W_R(a-round(.5*size(k,1)):a+round(1.5*size(k,1)),b-
round(.5*size(k,2)):b+round(1.5*size(k,2)))))<1 ))
            if (ibot(a+round(size(k,1)/2),b+2) >=FF26 |ibot(a+round(size(k,1)/2),b+3) >=FF26 ...
                |ibot(a+round(size(k,1)/2),b+1) >=FF26 )
                cc=cc+1;
            end
            if (ibot(a+round(size(k,1)/2),b+size(k,2)-2)>=FF26 |ibot(a+round(size(k,1)/2),b+size(k,2)-3)>=FF26 ...
                    |ibot(a+round(size(k,1)/2),b+size(k,2)-1)>=FF26 )
                cc=cc+1;
            end
            if (ibot(a+1,b+round(size(k,2)/2))>=FF26 | ibot(a,b+round(size(k,2)/2))>=FF26 ...
                    | ibot(a+2,b+round(size(k,2)/2))>=FF26)
                cc=cc+1;
            end
            if (ibot(a+size(k,1),b+round(size(k,2)/2))>=FF26 | ibot(a+size(k,1)-1,b+round(size(k,2)/2))>=FF26 ...
                | ibot(a+size(k,1)-2,b+round(size(k,2)/2))>=FF26)
                cc=cc+1;
            end
            if abs(sum(New(a:a+size(k,1),b)-
New(a:a+size(k,1),b+round(size(k,2)/2)))/sum((New(a:a+size(k,1),b)+New(a:a+size(k,1),b+round(size(k,2)/2)))) >
FF10
                cc=cc+1;
            end
            if cc>3
                W_R(a:a+size(K,1)-1,b:b+size(K,2)-1)=x1;
                W2(a:a+size(K,1)-1,b:b+size(K,2)-1)=1;
            end
        end

        b=b+1;
    end
    end
    a=a+1;
    b=round(1.5*size(k,2))+1;
    ii=1;
end

% Saving data
bead_locations{2}=[];
for ii=1:size(New,1)
    for jj=1:size(New,2)
        if W_R(ii,jj)==1
            bead_locations{1}(j1,1)=[ii];
            bead_locations{1}(j1,2)=[jj];
            j1=j1+1;
        end

        if W_W(ii,jj)==1
            bead_locations{2}(j2,1)=[ii];
            bead_locations{2}(j2,2)=[jj];
            j2=j2+1;
        end
    end
end
BL_RBC= sortrows(bead_locations{1});
```

```
BL_WBC= sortrows(bead_locations{2});


oo=size(Result_RBC,1);
for ot=1:size(BL_RBC,1)
        Result_RBC(ot+oo,1)=zz;
        Result_RBC(ot+oo,2)=mm;
        Result_RBC(ot+oo,3)=BL_RBC(ot,1);
        Result_RBC(ot+oo,4)=BL_RBC(ot,2);
end

ou=size(Result_WBC,1);
for op=1:size(BL_WBC,1)
        Result_WBC(op+ou,1)=zz;
        Result_WBC(op+ou,2)=mm;
        Result_WBC(op+ou,3)=BL_WBC(op,1);
        Result_WBC(op+ou,4)=BL_WBC(op,2);
end
% set_results{set_num}=Result_RBC;
cd processed

display(mm);
NewITitle=[];
WRTitle=[];
WWTitle=[];

NewITitle=[NewITitle; 'Cropped_New_', num2str(zz),'_',num2str(mm)];
WRTitle=[WRTitle; 'Cropped_WR_', num2str(zz),'_',num2str(mm)];
WWTitle=[WWTitle; 'Cropped_WW_', num2str(zz),'_',num2str(mm)];

imwrite(New,NewITitle, 'tif');
imwrite(W_R,WRTitle, 'tif');
imwrite(W_W,WWTitle, 'tif');
save result_RBC.txt Result_RBC -ascii -tabs;
save result_WBC.txt Result_WBC -ascii -tabs;
cd ..

i=[];
end


end
save result_RBC.txt Result_RBC -ascii -tabs;
save result_WBC.txt Result_WBC -ascii -tabs;
size(Result_RBC)
size(Result_WBC)
```