```
global lsm
%Test Image 1
%Original Image
OriginalSurface(:,:,1)=[15,15,15,180,180,150,150,150,150,150;
                  20,20,20,10,190,190,190,190,190,175;
                  20,20,20,10,180,180,190,185,180,185;
                  20,20,20,10,10,185,175,180,180,180];

OriginalSurface(:,:,2)=[10,185,10,20,20,20,10,20,20,20;
                  10,180,180,190,20,20,20,10,10,10;
                  10,180,185,185,20,190,190,185,20,20;
                  10,190,175,175,15,180,185,190,175,185];

OriginalSurface(:,:,3)=[20,20,10,10,10,20,20,20,20,20;
                  10,20,20,20,20,10,10,20,20,20;
                  10,185,20,20,20,20,20,20,30,30;
                  10,170,30,30,20,20,20,10,10,10];

OriginalSurface(:,:,4)=[10,10,10,20,40,20,20,20,30,40;
                  60,30,30,30,30,20,30,20,20,20;
                  30,30,50,50,50,30,30,30,30,30;
                  20,20,30,30,30,20,20,20,20,30];



OriginalBacteria(:,:,1)=[50,50,30,30,30,50,40,40,30,30;
                  30,40,20,20,20,30,50,50,30,70;
                  30,70,70,50,50,50,30,30,30,30;
                  30,40,40,30,20,20,20,60,50,50];

OriginalBacteria(:,:,2)=[40,40,40,185,30,30,30,190,40,40;
                  40,40,20,20,168,185,179,20,40,40;
                  40,20,10,20,190,10,10,10,145,155;
                  15,10,10,30,30,20,20,20,30,50];

OriginalBacteria(:,:,3)=[30,30,20,150,185,20,50,50,50,20;
                  180,130,20,145,20,30,30,30,50,50;
                  40,20,40,20,129,180,185,175,190,40;
                  10,40,40,20,175,20,40,175,20,20];

OriginalBacteria(:,:,4)=[40,30,40,180,175,50,40,40,30,40;
                  15,20,20,160,165,40,180,30,30,40;
                  15,40,20,185,190,40,190,160,40,40;
                  30,30,30,30,175,175,180,185,30,30];



%Image After Thresholding (Separate Biofilm Components From Background
%Noise)
Surface(:,:,1)=[0,0,0,1,1,1,1,1,1,1;
                  0,0,0,0,1,1,1,1,1,1;
                  0,0,0,0,1,1,1,1,1,1;
                  0,0,0,0,0,1,1,1,1,1];

Surface(:,:,2)=[0,1,0,0,0,0,0,0,0,0;
                  0,1,1,1,0,0,0,0,0,0;
                  0,1,1,1,0,1,1,1,0,0;
                  0,1,1,1,0,1,1,1,1,1];

Surface(:,:,3)=[0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0;
                  0,1,0,0,0,0,0,0,0,0;
                  0,1,0,0,0,0,0,0,0,0];
```

```
Surface(:,:,4)=[0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0];



Bacteria(:,:,1)=[0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0;
                  0,0,0,0,0,0,0,0,0,0];

Bacteria(:,:,2)=[0,0,0,1,0,0,0,1,0,0;
                  0,0,0,0,1,1,1,0,0,0;
                  0,0,0,0,1,0,0,0,1,1;
                  0,0,0,0,0,0,0,0,0,0];

Bacteria(:,:,3)=[0,0,0,1,1,0,0,0,0,0;
                  1,1,0,1,0,0,0,0,0,0;
                  0,0,0,0,1,1,1,1,1,0;
                  0,0,0,0,1,0,0,1,0,0];

Bacteria(:,:,4)=[0,0,0,1,1,0,0,0,0,0;
                  0,0,0,1,1,0,1,0,0,0;
                  0,0,0,1,1,0,1,1,0,0;
                  0,0,0,0,1,1,1,1,0,0];




pixelCol=10;
pixelRow=4;
totalpixel=pixelCol.*pixelRow;
%calibration to microns
micronx=0.5;
microny=0.5;
micronz=1;
%Number of image slices
yy=4;
lsm.yy=yy;



% Connected Volume Filtration (CVF) of the Bacteria and Surface
% Connected volume filtration code was provided open source by Dr. Arne Heydorn in his
% software program COMSTAT.  Permission was obtained on March 7, 2012, by Dr. Heydorn to
% use his CVF algorithm.  Source code can be obtained by Dr. Heydorn at
www.imageanalysis.dk.
%
% License for CVF
% Redistribution and use in source and binary forms, with or without
% modification, are permitted provided that the following conditions are met:
%
% 1. If the program is modified, redistributions must include a notice
% indicating that the redistributed program is not identical to the software
% distributed by the Department of Microbiology, Technical University of Denmark.
% Redistributions must also include a notice indicating that the redistributed
% program includes software developed by the Department of Microbiology,
% Technical University of Denmark.
%
% 2. All advertising materials mentioning features or use of this software must
% display the following acknowledgment: This product includes software developed
% by the Department of Microbiology, Technical University of Denmark.
```

```
%
% We also request that use of this software be cited in publications as
%
% Heydorn, A., Nielsen, A.T., Hentzer, M., Sternberg, C., Givskov, M., Ersbøll,
% B.K., Molin, S. (2000) Quantification of biofilm structures by the novel computer
% program COMSTAT. Microbiology 146 (10) 2395-2407
%
% The software is provided "AS-IS" and without warranty of any kind, express,
% implied or otherwise, including without limitation, any warranty of merchantability
% or fitness for a particular purpose. In no event shall the Technical University
% of Denmark or the authors be liable for any special, incidental, indirect or
% consequential damages of any kind, or any damages whatsoever resulting from loss
% of use, data or profits, whether or not advised of the possibility of damage, and
% on any theory of liability, arising out of or in connection with the use or
% performance of this software. This code was written using MATLAB 3.1 (MathWorks,
% www.mathworks.com) and may be subject to certain additional restrictions
% as a result.



%Modification of the CVF code
%April 2012:  Stacy Sommerfeld Ross modified the CVF from Dr. Heydorn to fit her
application
%and variables (see below).  This modified CVF (MCVF) is used to complete connected
%volume filtration on bacteria grown on uneven surfaces, in this case Surface.
%This code is provided "as is" without warranty.



%Connected Volume Filtration for Bacteria and Surface
BiofilmConnectedtoSurface=zeros(pixelRow, pixelCol, yy);

%Classifying Starting Point for Connected Volume Filtration
BiofilmConnectedtoSurface(:,:,1)=Bacteria(:,:,1); %This allows for any biofilm growing
directly on the substratum
HorizontalGrowthForBacteriaAttachedtoBStart(:,:,1)=Bacteria(:,:,1);

for i = 2:yy;
 commonarea(:,:,i-1)=uint8(Surface(:,:,i-1).*double((Bacteria(:,:,i)>0))); % what pixels
are common between the two layers
 [vectorx,vectory]=find(commonarea(:,:,i-1)); % vectors containing the nonzero elements
of commonarea
 BiofilmConnectedtoSurface(:,:,i)=commonarea(:,:,i-1); %  expand the area enherited from
the layer below

HorizontalGrowthForBacteriaAttachedtoBStart(:,:,i)=double(bwselect(Bacteria(:,:,i),vector
y,vectorx,8));
 clear vectorx
 clear vectory
end;

BiofilmConnectedtoSurfaceforSubstratum=BiofilmConnectedtoSurface;
for i=1:yy
    for Rowpixel=1:pixelRow;
            for Colpixel=1:pixelCol;
                if double(BiofilmConnectedtoSurface(Rowpixel,Colpixel,i))==1  &&
double(Surface(Rowpixel,Colpixel,i))==1
                    BiofilmConnectedtoSurfaceforSubstratum(Rowpixel,Colpixel,i)=0;
                elseif double(BiofilmConnectedtoSurface(Rowpixel,Colpixel,i))==1  &&
double(Surface(Rowpixel,Colpixel,i))==0
                    BiofilmConnectedtoSurfaceforSubstratum(Rowpixel,Colpixel,i)=1;
                elseif double(BiofilmConnectedtoSurface(Rowpixel,Colpixel,i))==0  &&
double(Surface(Rowpixel,Colpixel,i))==0
                    BiofilmConnectedtoSurfaceforSubstratum(Rowpixel,Colpixel,i)=0;
```

```
                elseif double(BiofilmConnectedtoSurface(Rowpixel,Colpixel,i))==0  &&
double(Surface(Rowpixel,Colpixel,i))==1
                    BiofilmConnectedtoSurfaceforSubstratum(Rowpixel,Colpixel,i)=0;
                end;
            end;
    end;
end;

%Modified Substratum Coverage
Substratum_Modified=sum(sum(sum(BiofilmConnectedtoSurfaceforSubstratum(:,:,:))))/(pixelCo
l*pixelRow)*100;

for i=1:yy
    for Rowpixel=1:pixelRow;
            for Colpixel=1:pixelCol;
                if double(Bacteria(Rowpixel,Colpixel,i))==1  &&
double(Surface(Rowpixel,Colpixel,i))==1
                    BacteriaColocalized(Rowpixel,Colpixel,i)=1;
                elseif double(Bacteria(Rowpixel,Colpixel,i))==1  &&
double(Surface(Rowpixel,Colpixel,i))==0
                    BacteriaColocalized(Rowpixel,Colpixel,i)=0;
                elseif double(Bacteria(Rowpixel,Colpixel,i))==0  &&
double(Surface(Rowpixel,Colpixel,i))==0
                    BacteriaColocalized(Rowpixel,Colpixel,i)=0;
                elseif double(Bacteria(Rowpixel,Colpixel,i))==0  &&
double(Surface(Rowpixel,Colpixel,i))==1
                    BacteriaColocalized(Rowpixel,Colpixel,i)=0;
                end;
            end;
    end;
end;

for i=1:yy

PercentBacteriaAssociatedwithSurface(i)=sum(sum(BacteriaColocalized(:,:,i)))/sum(sum(Bact
eria(:,:,i)))*100;
    NumberofBacteriaandSurfaceSamePixel(i)=sum(sum(BacteriaColocalized(:,:,i)));

VolumeofBacteriaandSurfaceSamePixel(i)=NumberofBacteriaandSurfaceSamePixel(i)*(micronx*mi
crony*micronz);
end;
PercentBacteriaAssociatedwithSurface=PercentBacteriaAssociatedwithSurface';
NumberofBacteriaandSurfaceSamePixel=NumberofBacteriaandSurfaceSamePixel';
VolumeofBacteriaandSurfaceSamePixel=VolumeofBacteriaandSurfaceSamePixel';
TotalBacteriaAssociatedwithSurface=sum(sum(sum(BacteriaColocalized)))/sum(sum(sum(Bacteri
a)))*100;
TotalNumberofBacteriaandSurfaceSamePixel=sum(sum(sum(BacteriaColocalized)));
TotalVolumeofBacteriaandSurfaceSamePixel=TotalNumberofBacteriaandSurfaceSamePixel*(micron
x*microny*micronz);
%Any "NAN - Not a real number" will be made 0.
PercentBacteriaAssociatedwithSurface(isnan(PercentBacteriaAssociatedwithSurface))=0;




for i=1:yy;
    for Rowpixel=1:pixelRow;
            for Colpixel=1:pixelCol;
                if double(BiofilmConnectedtoSurface(Rowpixel,Colpixel,i))==1  ||
double(HorizontalGrowthForBacteriaAttachedtoBStart(Rowpixel,Colpixel,i))==1
                    BasisforSubstratum(Rowpixel,Colpixel,i)=1;
```

```
                else
                    BasisforSubstratum(Rowpixel,Colpixel,i)=0;
                end;
            end;
    end;
end;

 %Determine the start points where the Biofilm Attaches to the Surface or
 %Substratum
  k=1;
  for i = 1:yy;
          if sum(sum(BiofilmConnectedtoSurface(:,:,i)))>=1
          StartIndex(k)=i;
          k=k+1;
          end;
   end;

   %Connect biofilm upward from each starting slice
Storage=zeros(pixelRow, pixelCol, yy);
NN=size(StartIndex);
for q=1:NN(2);
    SliceIndex=StartIndex(q);
    areainthislayer(:,:,1)=BasisforSubstratum(:,:,SliceIndex);

 for k = SliceIndex:yy-1;
        commonarea3=uint8(areainthislayer(:,:,k-
SliceIndex+1).*double((Bacteria(:,:,k+1)>0))); % what pixels are common between the two
layers
        [vectorx,vectory]=find(commonarea3); % vectors containing the nonzero elements of
commonarea
        %8 neighborhood connection for above slice
        %allows for horizontal growth of biofilm
        areainthislayer(:,:,k-
SliceIndex+2)=double(bwselect(Bacteria(:,:,k+1),vectory,vectorx,8));
        BiofilmAttachedtoBiofilm(:,:,k-SliceIndex+1)=areainthislayer(:,:,k-SliceIndex+2);

        %Keep track of where biofilm pixels grew
        for Rowpixel=1:pixelRow;
            for Colpixel=1:pixelCol;
                if BiofilmAttachedtoBiofilm(Rowpixel,Colpixel,k-SliceIndex+1)==1
                    Storage(Rowpixel, Colpixel, k+1)=1;
                end;
            end;
        end;

 end;


end;


%Put together the starting point BiofilmConnectedtoSurface and the
BiofilmAttachedtoBiofilm in Storage
filt_images=zeros(pixelRow,pixelCol,yy);
for i = 1:yy;
        for Rowpixel=1:pixelRow;
            for Colpixel=1:pixelCol;
```

```
                    if double(BiofilmConnectedtoSurface(Rowpixel,Colpixel,i)) ||
double(BasisforSubstratum(Rowpixel,Colpixel,i)) ==1 ||
double(Storage(Rowpixel,Colpixel,i)) ==1
                        filt_images(Rowpixel,Colpixel,i)=1;

                    end;
             end;
        end;
end;



%Quantification
UnconnectedBacteriaPixel=zeros(max(max(pixelRow)), max(max(pixelCol)), max(max(yy)));
ConnectedBiofilmBacteriaPixel=zeros(max(max(pixelRow)), max(max(pixelCol)), max(max(yy)));
AllSurfacePixels=double(Surface);
AllBacteriaPixels=double(Bacteria);

        for i=1:yy;
            for Rowpixel=1:pixelRow;
                for Colpixel=1:pixelCol;

                    if AllBacteriaPixels(Rowpixel, Colpixel,i)==1 && filt_images(Rowpixel,
Colpixel,i) ==1;
                        ConnectedBiofilmBacteriaPixel(Rowpixel, Colpixel,i)=1;
                        UnconnectedBacteriaPixel(Rowpixel, Colpixel,i)=0;
                    elseif AllBacteriaPixels(Rowpixel, Colpixel,i)==1 &&
filt_images(Rowpixel, Colpixel,i) ==0;
                        ConnectedBiofilmBacteriaPixel(Rowpixel, Colpixel,i)=0;
                        UnconnectedBacteriaPixel(Rowpixel, Colpixel,i)=1;
                    elseif AllBacteriaPixels(Rowpixel, Colpixel,i)==0 &&
filt_images(Rowpixel, Colpixel,i) ==0 || AllBacteriaPixels(Rowpixel, Colpixel,i)==0 &&
filt_images(Rowpixel, Colpixel,i) ==1;
                        ConnectedBiofilmBacteriaPixel(Rowpixel, Colpixel,i)=0;
                        UnconnectedBacteriaPixel(Rowpixel, Colpixel,i)=0;

                    end;

                end;
            end;
        end;



%Preallocate space
AreaConnectedBiofilmBacteriaBySlice=zeros( max(max(yy)), 1);
AreaUnconnectedBacteriaBySlice=zeros( max(max(yy)), 1);
AreaSurfacebySlice=zeros(max(max(yy)), 1);
AreaPSurfacebySlice=zeros(max(max(yy)), 1);
TotSurfaceBySlice=zeros(max(max(yy)), 1);
TotConnectedBiofilmBacteriaBySlice=zeros(max(max(yy)), 1);
TotUnconnectedBacteriaBySlice=zeros(max(max(yy)), 1);
AreaPercentConnectedBiofilmBacteriaBySlice=zeros(max(max(yy)), 1);
AreaPercentUnconnectedBacteriBySlice=zeros(max(max(yy)), 1);
TotalinMicrons=micronx*microny*pixelCol*pixelRow;

    %Surface
    AreaSurfaceOverall=sum(sum(sum(Surface(:,:,:))))*micronx*microny;
    AreaPSurfaceOverall=AreaSurfaceOverall/(TotalinMicrons*yy)*100;
    TotSurface=sum(sum(sum(Surface(:,:,:))));



    %Bacteria
```

```
AreaConnectedBiofilmBacteria=sum(sum(sum(ConnectedBiofilmBacteriaPixel(:,:,:))))*micronx*
microny;

AreaUnconnectedBacteria=sum(sum(sum(UnconnectedBacteriaPixel(:,:,:))))*micronx*microny;

AreaPercentConnectedBiofilmBacteria=AreaConnectedBiofilmBacteria/(TotalinMicrons*yy)*100;
    AreaPercentUnconnectedBacteria=AreaUnconnectedBacteria/(TotalinMicrons*yy)*100;
    TotConnectedBiofilmBacteria=sum(sum(sum(ConnectedBiofilmBacteriaPixel(:,:,:))));
    TotUnconnectedBacteria=sum(sum(sum(UnconnectedBacteriaPixel(:,:,:))));

    for i=1:yy;
    %Surface
    AreaSurfacebySlice(i)=sum(sum(Surface(:,:,i)))*micronx*microny;
    AreaPSurfacebySlice(i)=AreaSurfacebySlice(i)/TotalinMicrons*100;
    TotSurfaceBySlice(i)=sum(sum(sum(Surface(:,:,i))));



    %Bacteria

AreaConnectedBiofilmBacteriaBySlice(i)=sum(sum(ConnectedBiofilmBacteriaPixel(:,:,i)))*mic
ronx*microny;

AreaUnconnectedBacteriaBySlice(i)=sum(sum(UnconnectedBacteriaPixel(:,:,i)))*micronx*micro
ny;

AreaPercentConnectedBiofilmBacteriaBySlice(i)=AreaConnectedBiofilmBacteriaBySlice(i)/Tota
linMicrons*100;

AreaPercentUnconnectedBacteriBySlice(i)=AreaUnconnectedBacteriaBySlice(i)/TotalinMicrons*
100;

TotConnectedBiofilmBacteriaBySlice(i)=sum(sum(sum(ConnectedBiofilmBacteriaPixel(:,:,i))));
    TotUnconnectedBacteriaBySlice(i)=sum(sum(sum(UnconnectedBacteriaPixel(:,:,i))));

    end;


%Saving the Data
StackIndex=1;
lsm.slice=yy;
header={'Publication'};
filename=horzcat(num2str(StackIndex), '.xlsx');
colnames={'', 'Number of Connected Biofilm Pixels', 'Area Connected Biofilm, um^2', 'Area
Connected Biofilm, %',  'Number of Unconnected Bacteria Pixels', 'Area Unconnected
Bacteria, um^2', 'Area Unconnected Bacteria, %', 'Number of Surface Pixels','Area Surface,
um^2', 'Area Surface, %', 'Number Bacteria and Surface Pixels the Same', 'Volume Bacteria
and Surface Pixels the Same', 'Pecent Bacteria Associated with Surface, %', 'Modified
Substratum Coverage, %'};


%Biofilm Associated Bacteria
GA=vertcat(AreaConnectedBiofilmBacteria, AreaConnectedBiofilmBacteriaBySlice(:));
SubstratumSave=vertcat(Substratum_Modified,0);
TotalConnectedBiofilmBacteria=vertcat(TotConnectedBiofilmBacteria,TotConnectedBiofilmBact
eriaBySlice(:));
AreaPercentBio=vertcat(AreaPercentConnectedBiofilmBacteria,
AreaPercentConnectedBiofilmBacteriaBySlice(:));
AssociatedBacteria=vertcat(TotalBacteriaAssociatedwithSurface,
PercentBacteriaAssociatedwithSurface(:));
```

```
NumberAssociated=vertcat(TotalNumberofBacteriaandSurfaceSamePixel,NumberofBacteriaandSurf
aceSamePixel(:));
VolumeAssociated=vertcat(TotalVolumeofBacteriaandSurfaceSamePixel,VolumeofBacteriaandSurf
aceSamePixel(:));

%Non-Biofilm Associated Bacteria
FGA=vertcat(AreaUnconnectedBacteria,AreaUnconnectedBacteriaBySlice(:));
TotalUnconnectedBacteria=vertcat(TotUnconnectedBacteria,
TotUnconnectedBacteriaBySlice(:));
AreaPercentUnconnectedBacteriaCombined=vertcat(AreaPercentUnconnectedBacteria,
AreaPercentUnconnectedBacteriBySlice(:));

%Surface
TA=vertcat(AreaSurfaceOverall, AreaSurfacebySlice(:));
TotalSurface=vertcat(TotSurface, TotSurfaceBySlice(:));
SurfacePercent=vertcat( AreaPSurfaceOverall, AreaPSurfacebySlice(:));



%Thank you to Scott Hirsch (MathWorks, Natick, MA) for providing permission
% to use his xlswrite code in this application.
(http://www.mathworks.com/matlabcentral/fileexchange/2855-xlswrite)
%See xlswrite_BMT_Example.m for license information.
xlswrite_Image(StackIndex,TotalConnectedBiofilmBacteria, GA,
AreaPercentBio,TotalUnconnectedBacteria, FGA, AreaPercentUnconnectedBacteriaCombined,
TotalSurface, TA, SurfacePercent, NumberAssociated, VolumeAssociated, AssociatedBacteria,
SubstratumSave, header, colnames, filename);




%Generating ".tif" Sequences of Components
FinalConnectedBiofilmBacteria=zeros(max(max(pixelRow)), max(max(pixelCol)), max(max(yy)));
FinalUnconnectedBacteria=zeros(max(max(pixelRow)), max(max(pixelCol)), max(max(yy)));

folder=['Sample', int2str(StackIndex)];
parentfolderConnectedBiofilm=strcat(folder,'\ConnectedBiofilmBacteriaImages\');
parentfolderUnconnectedBacteria=strcat( folder,'\UnconnectedBacteriaImages\');
parentfolderOverall=strcat( folder,'\OverallBacteriaImages\');
parentfolderSurface=strcat( folder,'\SurfaceImages\');
parentfolderBacteriaSurface=strcat( folder,'\BacteriaAssociatedWithSurfaceImages\');
parentfolderBacteriaStars=strcat( folder,'\ModifiedSubstratumCoverageBacteria\');
mkdir(parentfolderConnectedBiofilm);
mkdir(parentfolderUnconnectedBacteria);
mkdir(parentfolderOverall);
mkdir(parentfolderSurface);
mkdir(parentfolderBacteriaSurface);
mkdir(parentfolderBacteriaStars);



       for i=1:yy;

FinalConnectedBiofilmBacteria(:,:,i)=ConnectedBiofilmBacteriaPixel(:,:,i).*double(Origina
lBacteria(:,:,i));
            name = ['ConnectedBiofilmBacteria',int2str(i),'.tif'];
            filename= strcat(parentfolderConnectedBiofilm, name);
                imwrite(uint8(FinalConnectedBiofilmBacteria(:,:,i)),filename,'tiff');

FinalUnconnectedBacteria(:,:,i)=UnconnectedBacteriaPixel(:,:,i).*double(OriginalBacteria(
:,:,i));
            name = ['UnconnectedBacteria',int2str(i),'.tif'];
            filename= strcat(parentfolderUnconnectedBacteria, name);
                imwrite(uint8(FinalUnconnectedBacteria(:,:,i)),filename,'tiff');
```

```
        end;

  FinalOverallBacteria=zeros(max(max(pixelRow)), max(max(pixelCol)), max(max(yy)));
  FinalSurface=zeros(max(max(pixelRow)), max(max(pixelCol)), max(max(yy)));
  FinalBacteriaAssociatedWithSurface=zeros(max(max(pixelRow)), max(max(pixelCol)),
max(max(yy)));

       for i=1:yy;

FinalOverallBacteria(:,:,i)=AllBacteriaPixels(:,:,i).*double(OriginalBacteria(:,:,i));
            name = ['OverallBacteria',int2str(i),'.tif'];
            filename= strcat(parentfolderOverall, name);
                imwrite(uint8(FinalOverallBacteria(:,:,i)),filename,'tiff');
            FinalSurface(:,:,i)=double(Surface(:,:,i)).*double(OriginalSurface(:,:,i));
            name = ['Surface',int2str(i),'.tif'];
            filename= strcat(parentfolderSurface, name);
                imwrite(uint8(FinalSurface(:,:,i)),filename,'tiff');

FinalBacteriaAssociatedWithSurface(:,:,i)=double(BacteriaColocalized(:,:,i)).*double(Orig
inalBacteria(:,:,i));
            name = ['BacteriaAssociatedWithSurface',int2str(i),'.tif'];
            filename= strcat(parentfolderBacteriaSurface, name);
                imwrite(uint8(FinalBacteriaAssociatedWithSurface(:,:,i)),filename,'tiff');

FinalBacteriaStars(:,:,i)=double(BiofilmConnectedtoSurface(:,:,i)).*double(OriginalBacter
ia(:,:,i));
            name = ['ModifiedSubstratumCoverageBacteria',int2str(i),'.tif'];
            filename= strcat(parentfolderBacteriaStars, name);
                imwrite(uint8(FinalBacteriaStars(:,:,i)),filename,'tiff');

       end;
```