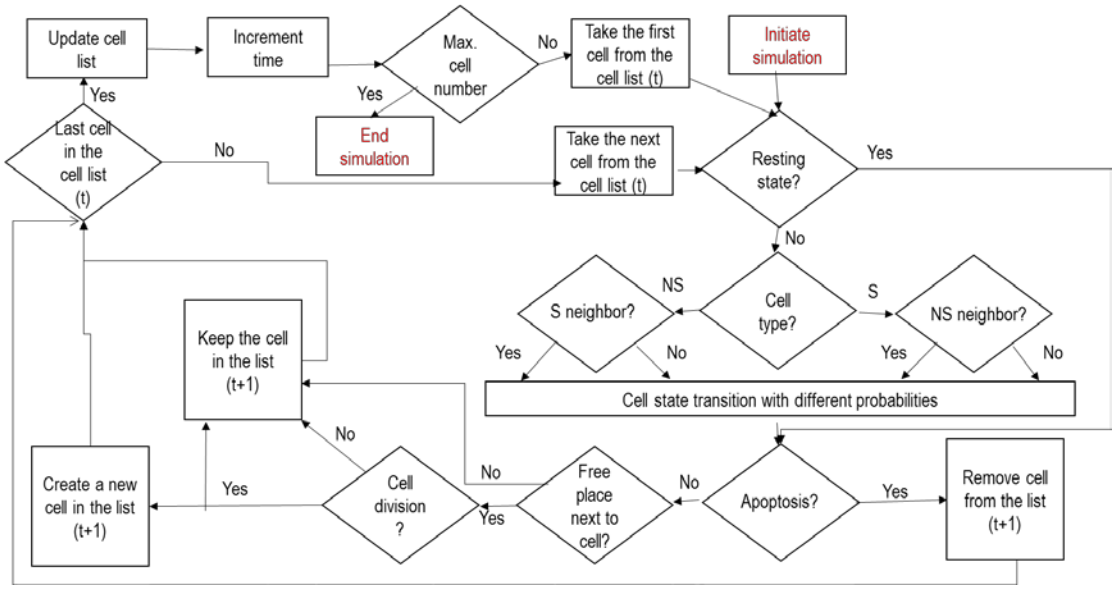
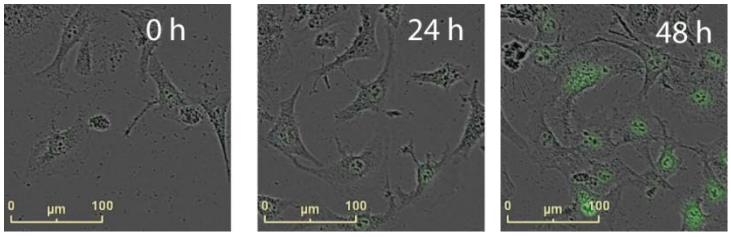
Supplementary Material section



Suppl. Figure S1 Flowchart of the simulation process. NS- bulk cell, S-stem cell



Suppl. Figure S2 No evidence for promoter silencing of the reporter construct after prolonged culturing of ZL55-SO^{high} cells A fraction of ZL55-SO^{high} cells (initially 100% eGFP(+)) lost eGFP expression after 2-month cultivation *in vitro*; a region enriched in eGFP(-) cells was selected (left). Cells were subsequently infected with a plasmid encoding the Yamanaka factor genes (*SOX2*, *OCT4*, *CMYC*, *KLF4*). 24 h post-infection (middle) and more clearly at 48 h (right), the appearance of green cells confirms the functionality of the initial reporter construct, i.e. to report on elevated levels of Sox2 and Oct4.

	Delta Ct	Delta Ct	Delta delta Ct	Expression fold change
Gene	ΔCt (ZL55-SO^{high})	ΔCt (ZL55-SO^{low})	ΔΔCt	2^ΔΔCt Fold increase in ZL55-SO^{high}
GLI1	9.3	9.0	0.2	1.2
gremlin	10.3	10.1	0.2	1.3

Supplementary Table 1. Primer pairs are the following: UBC forward 5'-GAT TTG GGT CGC GGT TCT T-3'; UBC reverse 5'-TGC CTT GAC ATT CTC GAT GGT-3' ; GLI1 forward 5'-CCA CAG TTA TGG GCC AGC CAG AGA G-3', GLI1 reverse 5'-GGC ATC CGA CAG AGG TGA GAT GGA C-3' ;Gremlin, forward, 5'-ATC AAC CGC TTC TGT TAC GG-3'; reverse, 5'-CGA TGG ATA TGC AAC GAC AC-3';

	Delta Ct	Delta Ct	Delta delta Ct	Expression fold change
Gene	ΔCt (ZL55-SO$^{clone-high}$)	ΔCt (ZL55-SO^{clonelow})	ΔΔCt	2^ΔΔCt Fold increase in ZL55-SO$^{clone-high}$
c-myc	8.6	8.4	0.2	1.2
ALDH1A1				
Sox2				
Oct4				

Supplementary Table 2. Primer pairs are the following: UBC forward 5'-GAT TTG GGT CGC GGT TCT T-3' ; UBC reverse 5'-TGC CTT GAC ATT CTC GAT GGT-3' ; c-myc forward 5'-GGA CTT GTT GCG GAA ACG AC-3' c-myc reverse 5'-CTC AGC CAA GGT TGT GAG GT-3' ; ALDH1A1 forward 5'- CTG CTG GCG ACA ATG GAG T-3' ALDH1A1 reverse 5'-GTC AGC CCA ACC TGC ACA G -3', Oct 4 forward 5'-GGG AGA TTG ATA ACT GGT GTG TT-3', Oct 4 reverse 5'-GTG TAT ATC CCA GGG TGA TCC TC-3' Sox 2 forward 5'-TAC AGC ATG TCC TAC TCG CAG-3' SOX2 reverse 5'-GAG GAA GAG GTA ACC ACA GGG-3'.

Supplementary Movie 1 Appearance of eGFP(+) stem cells from a single ZL55 bulk cell. Acquiring rate was 3h. Frame size is 400µmX400µm.

Supplementary Movie 2 Clustered appearance of eGFP(+) stem cells from a ZL55-SO^{low} population. Acquiring rate was 2h. Frame size is 400µmX400µm.

Supplementary Movie 3 Clustered appearance of eGFP(+) stem cells from a ZL55-SO^{low} population. Acquiring rate was 2h. Frame size is 400µmX400µm.

Supplementary Movie 4 Clustered appearance of eGFP(+) stem cells from a prMC-SO^{low} population. Acquiring rate was 2h. Frame size is 400µmX400µm.

Supplementary Movie 5 Clustered appearance of eGFP(+) stem cells from a prMC-SO^{low} population. Acquiring rate was 2h. Frame size is 400µmX400µm.

Supplementary Movie 6 Non-interacting Markov chain process does not produce clusters.

Supplementary Movie 7 Positively interacting Markov-chain process generates clusters.

Supplementary Movie 8 Tumor growth model initiated from one single stem cell

Supplementary Movie 9 Tumor growth model initiated from one single bulk cell

Supplementary MATLAB code 1

```
%% MATLAB CODE FOR ANALYSIS LOCALLY INTERACTING MARKOV-CHAIN
IN 25x25 GRID


close all
clear all

for q=1:1
clearvars -except keepVariables AnS q zScoreLS

gridsize=25
plate=zeros(gridsize, gridsize)

loc=zeros(gridsize*gridsize, 3)
initial=0 %1= Stem cell and 0=NonStem cell
plate(:,:)=initial
plateplus=plate;
loc(:, 3)= initial
runlength=2000
zscoreL=zeros(runlength,1)
nS=zeros(runlength,1)
T=zeros(runlength,1)
a=linspace(1, gridsize, gridsize)
b=linspace(1, gridsize, gridsize)

[X, Y]= meshgrid(a, b)
mesh=[X(:) Y(:)]

loc(:, 1)=mesh(:, 1)
loc(:, 2)=mesh(:, 2)
time=1

nCells=gridsize*gridsize
S=loc(:,3)
numberS=sum (S>0)
nS(time)=numberS
T(time)=time
vStoNS=0.009 %state transition probability
vNeighborStoNS=0.009 %state transition probability
vNStoS=0.001 %state transition probability
vNeighborNStoS=0.001 %state transition probability
imagetaken=10

%% Figure ini
lw = 2; % linewidth
fs = 8; % fontsize
ms = 2; % markersize
set(0,'DefaultFigureColor',[1,1,1]); % white background

g=figure (2)
ax=axes('Parent', g);
myCmap=[142/255 72/255 0 ; 0 1 0]
width = 2.5;
```

```
height = 2.5;
set(gcf,'Position',[280, 558, width*100, height*100]);
colormap(myCmap)
im=imagesc(plateplus,'Parent', ax)
caxis([0 1])



%% Loop


while time<runlength
    plate=plateplus;
    for i=1:nCells
        x=loc(i, 1);
    y=loc(i, 2);
    z=loc(i, 3);

    StoNS=rand();
    NStoS=rand();

    if x+1<=gridsize && x-1>0 && y+1<=gridsize &&  y-1>0 %No
border
        if (z==1 && StoNS<vStoNS && (plate(x+1, y)>0 &&
plate(x-1, y)>0 && plate(x, y+1)>0 &&  plate(x, y-1)>0))||
(z==1 && StoNS<vNeighborStoNS && (plate(x+1, y)==0 || plate(x-
1, y)==0 || plate(x, y+1)==0 ||  plate(x, y-1)==0))
             loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x+1, y)==0 &&
plate(x-1, y)==0 && plate(x, y+1)==0 &&  plate(x, y-1)==0))||
(z==0 && NStoS<vNeighborNStoS && (plate(x+1, y)>0 || plate(x-
1, y)>0 || plate(x, y+1)>0 ||  plate(x, y-1)>0))
               loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
    end

   if x-1==0 && y+1<=gridsize &&  y-1>0 %edge1
        if (z==1 && StoNS<vStoNS && (plate(x+1, y)>0 &&
plate(x, y+1)>0 &&  plate(x, y-1)>0)) || (z==1 &&
StoNS<vNeighborStoNS && (plate(x+1, y)==0 || plate(x, y+1)==0
||  plate(x, y-1)==0))
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x+1, y)==0 &&
plate(x, y+1)==0 &&  plate(x, y-1)==0)) || (z==0 &&
NStoS<vNeighborNStoS && (plate(x+1, y)>0 || plate(x, y+1)>0 ||
plate(x, y-1)>0))
        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
   end
```

```
   if x+1>gridsize && y+1<=gridsize &&  y-1>0 %edge2
        if (z==1 && StoNS<vStoNS && (plate(x-1, y)>0 &&
plate(x, y+1)>0 &&  plate(x, y-1)>0)) || (z==1 &&
StoNS<vNeighborStoNS && (plate(x-1, y)==0 || plate(x, y+1)==0
||  plate(x, y-1)==0))
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x-1, y)==0 &&
plate(x, y+1)==0 &&  plate(x, y-1)==0)) || (z==0 &&
NStoS<vNeighborNStoS && (plate(x-1, y)>0 || plate(x, y+1)>0 ||
plate(x, y-1)>0))
        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
   end

     if x+1<=gridsize && x-1>0 && y+1>gridsize %edge3
        if (z==1 && StoNS<vStoNS && (plate(x+1, y)>0 &&
plate(x-1, y)>0 && plate(x, y-1)>0 )) || (z==1 &&
StoNS<vNeighborStoNS && (plate(x+1, y)==0 || plate(x-1, y)==0
|| plate(x, y-1)==0 ))
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x+1, y)==0 &&
plate(x-1, y)==0 && plate(x, y-1)==0 )) || (z==0 &&
NStoS<vNeighborNStoS && (plate(x+1, y)>0 || plate(x-1, y)>0 ||
plate(x, y-1)>0 ))
        z=1
        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
     end

    if x+1<=gridsize && x-1>0 && y-1==0 %edge4
        if (z==1 && StoNS<vStoNS && (plate(x+1, y)>0 &&
plate(x-1, y)>0 && plate(x, y+1)>0 )) || (z==1 &&
StoNS<vNeighborStoNS && (plate(x+1, y)==0 || plate(x-1, y)==0
|| plate(x, y+1)==0 ))
                loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x+1, y)==0 &&
plate(x-1, y)==0 && plate(x, y+1)==0 )) || (z==0 &&
NStoS<vNeighborNStoS && (plate(x+1, y)>0 || plate(x-1, y)>0 ||
plate(x, y+1)>0 ))
        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
    end

    if x-1==0 && y+1>gridsize %corner1
        if (z==1 && StoNS<vStoNS && (plate(x+1, y)>0 &&
plate(x, y-1)>0)) || (z==1 && StoNS<vNeighborStoNS &&
(plate(x+1, y)==0 ||  plate(x, y-1)==0))
```

```
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x+1, y)==0 &&
plate(x, y-1)==0)) || (z==0 && NStoS<vNeighborNStoS &&
(plate(x+1, y)>0 ||  plate(x, y-1)>0))

        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
   end

   if x+1>gridsize &&  y+1>gridsize %corner2
        if (z==1 && StoNS<vStoNS && (plate(x-1, y)>0 &&
plate(x, y-1)>0)) || (z==1 && StoNS<vNeighborStoNS &&
(plate(x-1, y)==0 ||  plate(x, y-1)==0))
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x-1, y)==0 &&
plate(x, y-1)==0)) || (z==0 && NStoS<vNeighborNStoS &&
(plate(x-1, y)>0 ||  plate(x, y-1)>0))
        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
   end

     if x-1==0 && y-1==0 %corner3
        if (z==1 && StoNS<vStoNS && (plate(x+1, y)>0  &&
plate(x, y+1)>0 )) || (z==1 && StoNS<vNeighborStoNS &&
(plate(x+1, y)==0  || plate(x, y+1)==0 ))
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x+1, y)==0  &&
plate(x, y+1)==0 )) || (z==0 && NStoS<vNeighborNStoS &&
(plate(x+1, y)>0  || plate(x, y+1)>0 ))
               loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
     end

    if x+1>gridsize && y-1==0 %corner4
        if (z==1 && StoNS<vStoNS && (plate(x-1, y)>0 &&
plate(x, y+1)>0 )) || (z==1 && StoNS<vNeighborStoNS &&
(plate(x-1, y)==0 || plate(x, y+1)==0 ))
        loc(i, :, :) = [x, y, 0];
        plateplus(x, y)=0;
        end
         if (z==0 && NStoS<vNStoS && (plate(x-1, y)==0 &&
plate(x, y+1)==0 )) || (z==0 && NStoS<vNeighborNStoS &&
(plate(x-1, y)>0 || plate(x, y+1)>0 ))
        loc(i, :, :) = [x, y, 1];
        plateplus(x, y)=1;
        end
     end
```

```
    end
    time=time+1;
             if mod(time,(imagetaken/time))==0 || time==1

    %

  figure (2)
    ax=axes('Parent', g);
set(gcf,'Position',[280, 558, width*100, height*100]);
myCmap=[142/255 72/255 0 ; 0 1 0];
colormap(myCmap);
im=imagesc(plateplus,'Parent', ax);
caxis([0 1]);
    %title(['gremlin=' num2str(time) ]);

       filename = sprintf('B:PLMD.png',time);
        saveas(g, filename);

end

S=loc(:,3);
numberS=sum (S>0);% counting the number of S
nS(time)=numberS;
T(time)=time;

%Neirest neighbor index calculation

XX=loc(loc(:,3)>0,:); %select the green cells
cloud=(XX(:,1:2));
Lcloud=size(cloud,1);
 zscore=0;
if Lcloud>3
    posS=zeros(1, Lcloud);
    posl=zeros(1, Lcloud);
    %posN=[];
    sor=1;
    m1=[];
    for jj=1:Lcloud
    first= cloud (jj,1);
    second=cloud (jj,2);
        if first==0 || first==gridsize || second==0 ||
second==gridsize
            posS(jj)=jj;
            m1(sor)=jj;
            sor=sor+1;
        end
    posl(jj)=jj;
    end
    posN=posl-posS;
    posN=posN(posN>0);
    cloudnoborder=cloud;
    cloudnoborder(m1,:)=[];
    Lcloudnoborder=size(cloudnoborder,1);

    for i=1:Lcloudnoborder
```

```
        query=cloudnoborder(i,:);
        reduced=cloud;
        xxx=posN(i);
        reduced(xxx, :) = [];
        [IDX,D] = knnsearch(reduced, query);
        DlistB(i)=D;
    end
        meanDlist=mean(DlistB);
    NNI=2*meanDlist*sqrt(Lcloud/((gridsize+1)*(gridsize+1))) ;
    AperN=((gridsize+1)*(gridsize+1))/Lcloud;
expected=0.5*sqrt(AperN);
SE= 0.26136/(sqrt(Lcloud^2/((gridsize+1)*(gridsize+1))));
   zscore=(meanDlist-expected)/SE;%
end
 zscoreL(time)=zscore;
end
AnS(:,q)=nS
zScoreLS(:, q)=zscoreL'
end


filename =
'B:\Laci\article\CSCmodel\JBC\revision\Regular.xlsx';
sheet = 2
xlswrite(filename,AnS)
xlswrite(filename,zScoreLS, 2)
```

Supplementary MATLAB code 2

```
% TUMOR GROWTH MODEL
% Tumor growth model considering the stem cell state of the the
individual cancer cells.
% 0 == No cancer cell, white
% 1 == Tumor bulk cell, brown
% 2 == Stem cell, green
%
close all
clear all


size=100;% size of the matrix
plate=zeros(size, size);% cell 2D matrix
nCells=1; %initial number of cell
A=0; % cell death number
loc=zeros(size*size, 4);
x=round(size/2); % the place of the initial cell
y=round(size/2); % the place of the initial cell
stem=1 ;%initial cell 2=stem cell, 1=bulk cell
```

```
g=1 ; % counts the life time of a cell
loc(nCells,:,:)=[x, y, stem, g]; % list of cells
plate (x, y)=stem ;
plateplus=plate;
time=1;
T(time)=time;
S=loc(:,3);
numberS=sum(S>1); % counts the number of stemcells
Stemnumber(time)=numberS;
Totalcellnumber(time)=nCells;
Sratio(time)=numberS/nCells ; % counts the ratio of stemcells

f=figure;
a=axes('Parent', f);
 myCmap = [1 1 1  ; 142/255 72/255 0   ;  0 1 0];
 colormap(myCmap);
im=imagesc(plateplus, 'Parent', a);
caxis([0 2]);
colorbar;
% VIDEO INITIALIZATION
    writerObj = VideoWriter('B:TumorgrowthNS.avi')
    writerObj.FrameRate = 4;  % Default 30
    open(writerObj);

    hFig=f;
    set(hFig, 'Position', [0 0 1200 900]);
 frame = getframe(gcf);
        writeVideo(writerObj,frame);





while(nCells <=2000) % maximal number of cells
    A=0 ;%number of cell death
    plate=plateplus;
    nCellsStop=nCells; % to exclude the cells generated in the cycle
from the re-cycling
     for         i=1:nCellsStop
           x=loc(i,1);
           y=loc(i,2);
```

```
         stem=loc(i, 3);
         g=loc(i, 4);
        sk=rand(); %random number generation to become a stem cell
        sr=rand() ;%random number generation to become a bulkcell
from stem cell
        g=g+1; % lifetime of a cell
        loc (i,:,:,:)=[x,y, stem, g];

        %dediferentiation
        if (sk<0.0018 && stem==1 && g<24 && (plate(x+1,y)<2 &&
plate(x-1,y)<2 && plate(x, y-1)<2 && plate(x,y+1)<2)) ||...
                (sk<0.06 && stem==1 && g<24 && (plate(x+1,y)==2 ||
plate(x-1,y)==2 || plate(x, y-1)==2||plate(x,y+1)==2)); %first No
neighbour 2nd there is a neighbour
              stem=stem+1;

              loc(i,:,:,:)=[x, y, stem,g];
              plateplus(x,y)=stem;
        end

        %diferentiation
        if (sr<0.03 && stem==2 && g<24 && (plate(x+1,y)~=1 &&
plate(x-1,y)~=1 && plate(x, y-1)~=1 && plate(x,y+1)~=1)) ||...
                (sr<0.06 && stem==2 && g<24 && (plate(x+1,y)==1 ||
plate(x-1,y)==1 || plate(x, y-1)==1 || plate(x,y+1)==1)); %first No
neighbour 2nd there is a neighbour
              stem=stem-1;

              loc(i,:,:,:)=[x, y, stem, g];
              plateplus(x,y)=stem;
        end
   %CELL DIVISION
                          divide=rand();  %random number generation
for cell division or cell death

        if (divide<=0.15 && (plateplus(x+1,y)==0 || plateplus(x-
1,y)==0 || plateplus(x, y-1)==0||plateplus(x,y+1)==0)  && stem==1 )
||...
```

```
                (divide<=0.07 && (plateplus(x+1,y)==0 || plateplus(x-
1,y)==0 || plateplus(x, y-1)==0||plateplus(x,y+1)==0)  && stem==2
);%cell division probanilities
            nCells=nCells+1 ;
           divideDir=randi(4);
           x=loc(i,1);
           y=loc(i,2);
           stem=loc(i,3);
           g=loc(i,4);
        %find a place for the new cell
           if divideDir==1 && plateplus(x+1,y)==0;
           x=x+1;
           else
               if divideDir==1 && plateplus(x-1,y)==0  ;
                   x=x-1;
                   else
                   if divideDir==1 && plateplus(x, y-1)==0 ;
                      y=y-1;
                   else
                       if divideDir==1 && plateplus(x,y+1)==0  ;
                       y=y+1;
                       end
                   end
               end
           end

          if divideDir==2 && plateplus(x-1,y)==0;
           x=x-1;
           else
               if divideDir==2 && plateplus(x,y-1)==0  ;
                   y=y-1;
                   else
                   if divideDir==2 && plateplus(x, y+1)==0 ;
                      y=y+1;
                    else
                       if divideDir==2 && plateplus(x+1,y)==0  ;
                       x=x+1;
                      end
                   end
               end
          end
```

```
 if divideDir==3 && plateplus(x,y-1)==0;
y=y-1;
    else
    if divideDir==3 && plateplus(x,y+1)==0  ;
        y=y+1;
      else
        if divideDir==3 && plateplus(x+1, y)==0 ;
           x=x+1;
         else
            if divideDir==3 && plateplus(x-1,y)==0 ;
            x=x-1;
            end
        end
    end
 end

if divideDir==4 && plateplus(x,y+1)==0;
y=y+1;
else
    if divideDir==4 && plateplus(x+1,y)==0  ;
        x=x+1;
        else
        if divideDir==4 && plateplus(x-1, y)==0 ;
           x=x-1;
        else
            if divideDir==4 && plateplus(x,y-1)==0  ;
            y=y-1;
            end
        end
    end
 end


loc(nCells,:,:,:)=[x, y,stem, 1];
 plateplus(x, y)=stem;
caxis([0 2]);
colorbar
        end
```

```
          if divide>0.998 && time>4 && stem==1  || divide>0.999 &&
time>4 && stem==2 % cell death probability
              loc(i,:,:,:)=[0 0 0 0];
plateplus(x, y)=0;
            A=A+1; %apoptotic cell counter

         end

     end
     loc=sortrows(loc, -3);
     nCells=nCells-A%subract the death cell number
     time=time+1
     set (im,'CData',plateplus);

     T(time)=time;
     Totalcellnumber(time)=nCells;
     S=loc(:,3);
numberS=sum(S>1);
Stemnumber(time)=numberS;
Sratio(time)=numberS/nCells;
        pause(0.001)
% write video object from current frame
 hFig=f;
 set(hFig, 'Position', [0 0 1200 900])  ;
 frame = getframe(gcf);
        writeVideo(writerObj,frame);
end
%movie2avi(M, 'MOVE15trial.avi','fps',7,'compression','None')
 % close and save video object
     close(writerObj);



plot(T,Sratio)
axis([0 time 0 1])
xlabel('Time')
ylabel('CSC/(CSC+NCSC)')
print ('Plot11','-dtiff')

plot(T,Totalcellnumber,'r', T, Stemnumber, 'g')
```