Supporting Information for

Development of Surface Functionalization Strategies for 3D-Printed Polystyrene Constructs

Max J. Lerman[1,2,3], Shin Muramoto[2], Navein Arumugasaamy[3,4,5], Michael Van Order[1], Josephine Lembong[3,4], Anushka Gerald[3,4], Greg Gillen[2], John P. Fisher[3,4*]

[1]Department of Materials Science and Engineering, University of Maryland, College Park MD
[2]Surface and Trace Chemical Analysis Group, Materials Measurement Lab, National Institute of Standards and Technology, Gaithersburg, MD
[3]Center for Engineering Complex Tissues, University of Maryland, College Park, MD
[4]Fischell Department of Bioengineering, University of Maryland, College Park MD
[5]Sheikh Zayed Institute for Pediatric Surgical Innovation, Children's National Health System, Washington, DC

Appendix 1: Image Analysis Program and documentation

```
% ImageProcessingCells
% Reads in images, reads in size of images
% Sorts by stack location
% Opens an image of the cell dots (indicating the % cells locations)
% Color separations
% Traces outlines, displays areas of cells and statistics of stack
%Assumes 1024x1024 images

% Read in directory of images
% Get list of all TIFF files in this directory
% DIR returns as a structure array.

imagefiles = dir('*.tif');
nfiles = length(imagefiles);    % Number of files found
images = cell(nfiles,1);
disp('Files Found')
for ii = 1:nfiles
   currentfilename = imagefiles(ii).name;
   currentimage = imread(currentfilename);
   images{ii} = currentimage;
end
disp('Files Read')
% Sort stacks
% All stacks will have a known number of images in stack
% e.g. a 4x4 stack will have 16 separate groups of images, separate by
% number of images and get maximum image projection for each group
rows = str2double(input('How many rows in set? ', 's'));
cols = str2double(input('How many columns in set? ', 's'));
pxl = str2double(input('What is the image resolution (pixels/um default 1.6516)? ','s'));
thu = str2double(input('What is your threshold area (um^2)? ', 's'));

grps = cell(rows*cols,1);
ngrps = length(grps);

% Convert threshold to pixels
thp = floor((pxl^2)*thu);

%Sort images, and read into maximum images for each images set
%Define variables for speed
zStack = nfiles/ngrps;
jj = 1;
grpsr = cell(rows*cols,1);
grpsb = cell(rows*cols,1);
grpsrt = cell(rows*cols,1);
grpsbt = cell(rows*cols,1);
```

```
hld = zeros(1024,1024,3);
maxImageHoldr = zeros(1024,1024,zStack);
maxImageHoldb = zeros(1024,1024,zStack);
cellarea = zeros(rows*cols,1);

for ii = 1:ngrps
    for kk = 1:zStack
        % Separate Colors (Blue, Green, Red)
        hld = images{jj};
        thr = graythresh(hld(:,:,1));
        thb = graythresh(hld(:,:,3));

        % Threshold out objects below the intensity threshold
        % Tuned based on individual images. 0-0.07 Typical.
        if thr < 0.000
            thr = 0.000;
        end

        if thb < 0.000
            thb = 0.000;
        end

        maxImageHoldr(:,:,kk) = imbinarize(hld(:,:,1),thr);
        maxImageHoldb(:,:,kk) = imbinarize(hld(:,:,3),thb);

        jj = jj + 1;

    end
    fprintf('Separated\n')
    % Maximum Image Projection
    grpsr{ii} = max(maxImageHoldr,[],3);
    grpsb{ii} = max(maxImageHoldb,[],3);

    % Threshold out objects below the area threshold
    grpsrt{ii} = bwareaopen(grpsr{ii},thp);
    grpsbt{ii} = bwareaopen(grpsb{ii},thp);
    fprintf('Projected and Thresholded\n')

    figure(ii)
    warning off
    imshowpair(grpsrt{ii},grpsbt{ii},'montage')
    warning on

    % Pull area information from each captured portion
    % First determine connected components in the images
    cr = bwconncomp(grpsrt{ii});
```

```
    cb = bwconncomp(grpsbt{ii});

    % Find the area of those connected components
    ar = regionprops(cr,'Area');
    ab = regionprops(cb,'Area');



    % Number of detected nucleii
    nnucleii = length(ab);
    naru = length(ar);
    cella = 0;
    aru = cell(naru,1);
    abu = cell(nnucleii,1);

    for pp = 1:naru

        %Convert to um^2
        aru{pp} = ar(pp).Area./(pxl^2);

        % Area covered by Actin
        cella = cella + aru{pp};
    end


    %Vector computed containing the average area coverage in each stack
    cellarea(ii) = cella/nnucleii;


    fprintf('Layer %d of %d completed, area computed\n',ii,ngrps)
end


% Calculate statistics of stack
marea = mean(cellarea);
stdarea = std(cellarea);

fprintf('The mean of the stack is %d +/- %d.\n',marea,stdarea)
```

Table S1. XPS compositional analysis of TCPS and NT (control), as well as PS surfaces treated with He, ACN, and O_2 plasma. A Kratos Axis Ultra delay-line detector, (Kratos Analytical LtD, Manchester, United Kingdom) used in hybrid mode with a monochromatic Al Kα1, 2 X-ray source (hv = 1486.6 eV). The survey scan shows the elemental distribution of the surface in percent atomic composition, while the high resolution scan shows the chemical states for the elements carbon and nitrogen. Data is shown as mean ± standard deviation

	Survey Scan			High Resolution Scan				
	O 1s	C 1s	N 1s	285.3 C-C	286.4 C-O	287.6 C=O	397.9 N-R*	402.4 Malo**
NT	0 ± 1	99 ± 1	-	100	-	-	-	-
ACN	8 ± 1	63 ± 1	29 ± 1	27 ± 1	52 ± 2	24 ± 2	89 ± 7	11 ± 7
TCPS	12 ± 1	88 ± 1	0 ± 1	84 ± 1	13 ± 1	3 ± 1	-	-
O_2	11 ± 1	89 ± 1	1 ± 1	82 ± 2	12 ± 4	6 ± 5	-	-
He	10 ± 1	89 ± 1	1 ± 1	85 ± 2	8 ± 2	7 ± 2	-	-

*Imine (R=N-R) where R is aromatic

**Malononitrile (N≡C-CH2-C≡N)

Table S2. Area fractions of each molecular weight peak was calculated and used to determine relative quantity of each peak. No difference between the samples was observed ($p = 0.905$)

	Stock (%)	Printed (%)
High Fraction	33.5 ± 0.2	33.5 ± 0.5
Low Fraction	66.5 ± 0.2	66.5 ± 0.5