

## Supplementary information

### S 1. Mobile application for colorimetric measurement

The mobile application can process images captured using the phone camera as well as previously saved photos. The source code for the application was included in the supplement (Table S3). Although this mobile application was written using a Windows Phone 7 operating system, the cell phone code algorithm can be applicable to various other smart phone platforms.

This mobile application was designed to pick image regions that represent the color intensity within microchannels, to average pixel values therein (red channel) and to report the biomarker concentration. To automatically select image regions for analysis, a basic search algorithm was employed. The algorithm was designed to convert the color intensity into red, green and blue pixel values and to select the regions that had the highest pixel values below an R color threshold (a default value of 70). This threshold was based on the clinical images obtained, and was implemented as a user-defined modifiable parameter. Briefly, the algorithm first checks whether the region starting from the selected pixel is a continuous region with low red values. Otherwise, it offsets the region until a region is found, which has the R value below the threshold (70). Some of the microchannel images have low color intensity, and do not provide clear difference from the background. To facilitate region selection by the mobile application, those images were modified to add small markers next to microchannels to assist the selection of detection region. In the future versions of the microchips, markers can be placed on the microchips to facilitate taking images and image recognition algorithms. The mobile application assumes that the captured images were oriented horizontally with small rotation angles. As such, the selected regions were axis aligned rectangles and each region in the same image was vertically aligned.

Following the selection of the image region, as indicated by the red rectangle (Figure 1D), the mobile application was designed to convert the color intensity to RGB pixel values. To minimize the illumination difference from chip-to-chip, we normalized the color intensity based on the difference in the background between the standard and sample images. The application selected a typical background region from each sample image and compared the R values therein to an average of R values from the background regions in standard images. The R values from the selected region in sample images were offset by deducing the R value difference. After normalization, the pixel values exported from standard images were used to generate the standard curve, by which the concentration of HE4 in sample images was calculated and reported. The mobile application also reported R values obtained from the automatically selected regions within the microchannels and also the concentration of HE4 (ng/mL) on the cell phone screen (Figure 1D).

## **S 2. Mobile Application Instructions**

To execute the mobile application, a phone with Windows Phone 7 operating system or Windows Phone 7 emulator on a PC can be used. Windows Phone 7 Emulator is included in free Windows Phone Developer Tools download at <http://create.msdn.com/en-US/resources/downloads> (accessed on March 18<sup>th</sup>, 2011). To install the application, 'HE4Detect.xap' was deployed using 'Application Deployment' tool that comes with Windows Phone Developer tools.

The main page of the mobile application allows users to choose and analyze an image. Users can choose an image by taking a photo using the mobile camera or selecting an image loaded onto

the cell phone. To calculate the standard curve using standard images, click settings and check 'standard curve calibration'. Standard images must to be loaded in an order starting from higher concentrations to lower ones for regions to be assigned to the right concentration values. After loading the images, the mobile application calculates R values and updates standard curve regression parameters. To analyze the clinical testing images, click settings and uncheck 'standard curve calibration'. The application will report R values and concentrations for each selected image.

### S 3. Matlab code for colorimetric measurement

A MATLAB code was used to analyze the images taken by a CCD camera or a cell phone. We also used the detected R values obtained by the MATLAB as a control to validate the development of the mobile application. Since images were saved in jpeg format by the cell phone, the compression algorithm causes negligible R value differences within less than 1%. The MATLAB code is shown below.

```
for i=1:34
str = strcat(int2str(i),'.jpeg');
I=imread(str);
Ired = I;
Igreen = I;
Iblue = I;
Ired(:,:,2)=[];
Ired(:,:,2)=[];
Igreen(:,:,1)=[];
Igreen(:,:,2)=[];
Iblue(:,:,1)=[];
Iblue(:,:,1)=[];
r=mean2(Ired);
g=mean2(Igreen);
b=mean2(Iblue);
std=std2(Iblue);
rgb(i,1)=r;
rgb(i,2)=g;
rgb(i,3)=b;
rgb(i,4)= std;
end
rgb
xlswrite('rgb.xls',rgb);
```

**Table S1. Comparison of Red pixel values obtained by MATLAB and mobile application.**

<b>HE4 Concentration (ng/mL)</b>	<b>Average R values by MATLAB</b>	<b>Average R values by mobile application</b>
1250.0	2.662	2.165
625.0	14.737	14.708
312.5	35.597	35.610
156.3	58.823	57.865
78.1	69.038	68.213
39.1	82.520	81.805
19.5	93.251	92.723
0.0	121.014	121.090

HE4 peptide standards were tested by microchip ELISA. The image results taken using a cell phone were analyzed by MATLAB and the developed mobile application. Red (R) pixel values obtained by MATLAB were used a control method to validate the development of the mobile application.

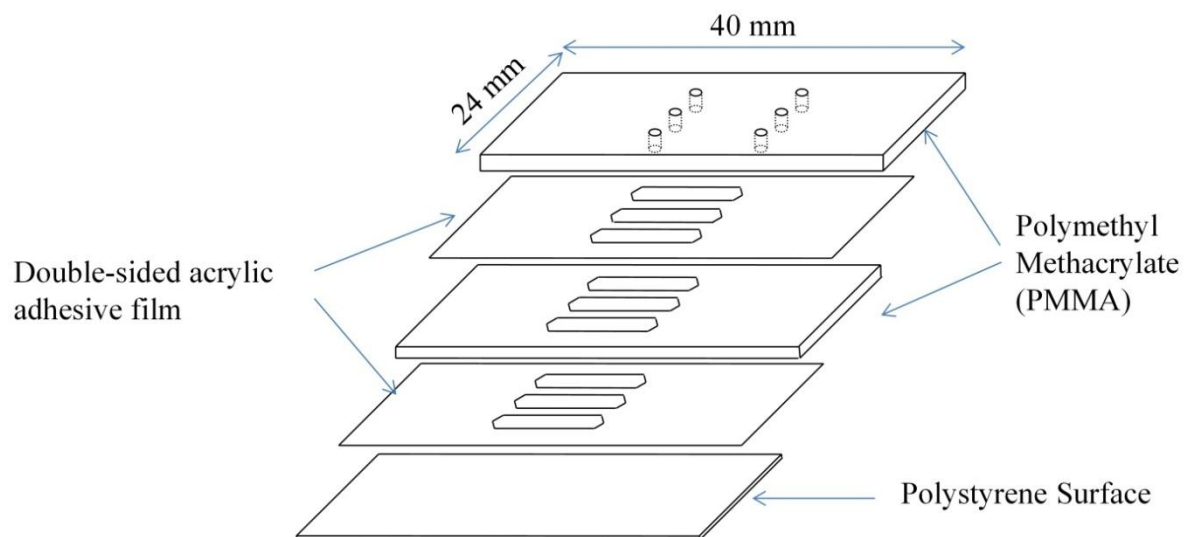
**Table S2. Reproducibility of microchip ELISA coupled with cell phone detection.**

	Linearity ( $R^2$ )
Experiment 1	0.938
Experiment 2	0.992
Experiment 3	0.972

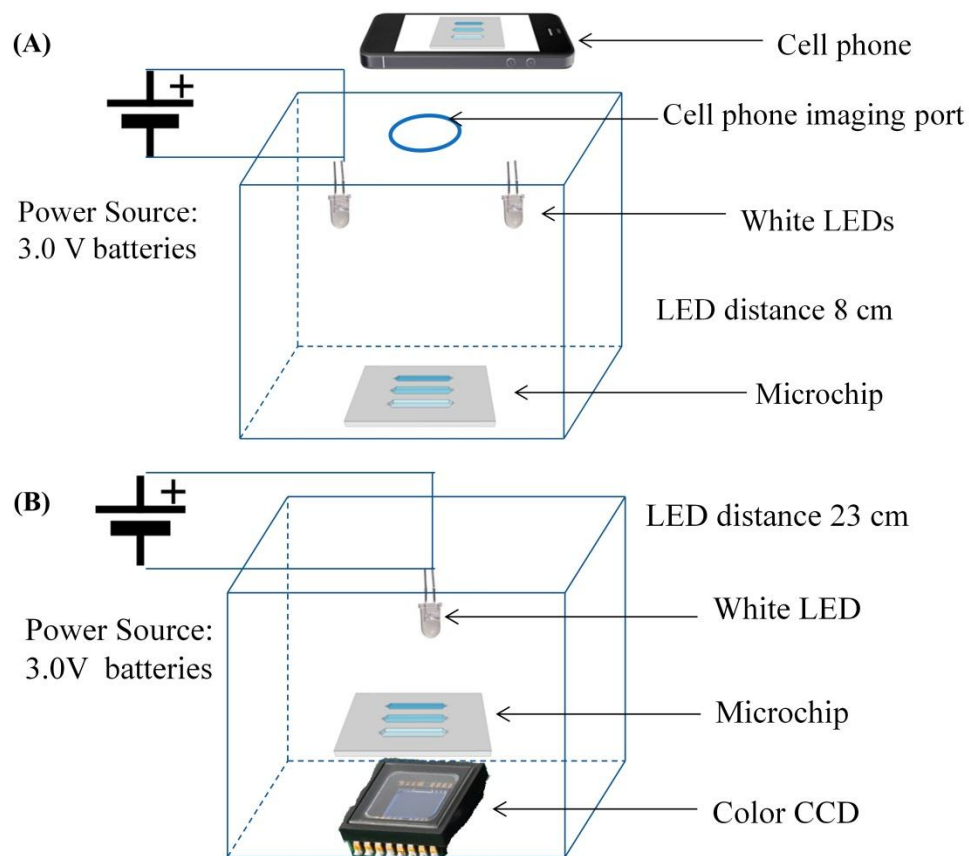
Three independent experiments were evaluated for the reproducibility. In these experiments, replicates of the HE4 standards ranging from 1,250, 625.0, 312.5, 156.3, 78.1, 39.1 and 19.5 ng/mL were included in microchip ELISA coupled with cell phone detection.

**Table S3. Mobile application code files to perform colorimetric detection for microchip ELISA**

File	Purpose
MainPage.xaml.cs	Main code to handle UI changes and analyze images
MainPage.xaml	UI for the main page of the application
App.xaml.cs	Code to handle application events
AppSettings.cs	Code to manage application settings
HE4Detect.sln	Visual Studio solution file for the mobile application
HE4Detect.csproj	Visual Studio project file for the mobile application
HE4Detect.xap	Deployment file to install the mobile application

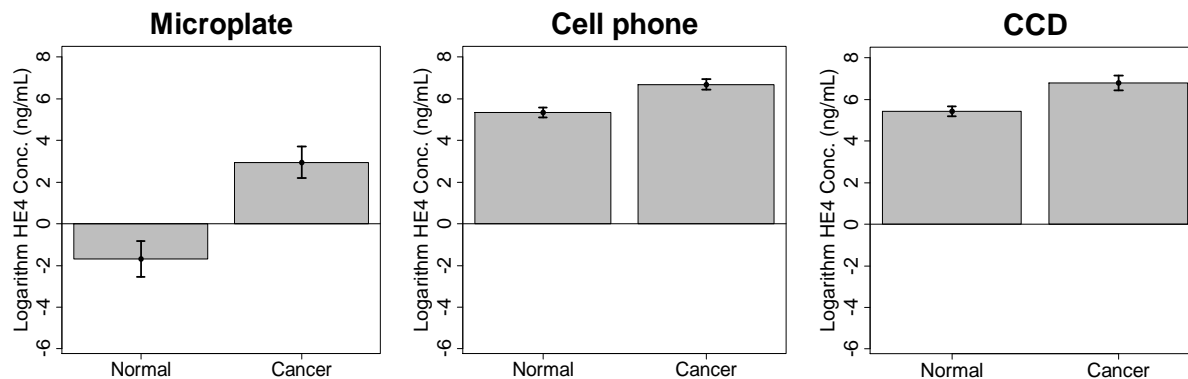


**Figure S1. Layer-by-layer assembly of an ELISA microchip.**



**Figure S2. Colorimetric measurement setups for a cell phone (A) and a lensless CCD (B).**





**Figure S3. Barplot of mean logarithm-transformed HE4 concentration with 95% confidence interval.**

For all three methods, the 95% confidence interval (CI) for mean logarithm-transformed HE4 concentration for cancer group (n=19) was well above that for normal group (n=20). This indicated the logarithm-transformed HE4 concentration for majority of cancer urine samples was much greater than that of normal urine samples. For the microplate method, the means, standard errors of the sample mean (SEMs), and 95% CIs are -1.69, 0.31, [-2.29, -1.08] for normal urine samples and were 2.95, 0.27, [2.42, 3.47] for cancer urine samples. For the cell phone method, the means, SEMs, 95% CIs were 5.35, 0.09, [5.17, 5.52] for normal urine samples and were 6.68, 0.09, [6.50, 6.86] for cancer urine samples. For the CCD method, the means, SEMs and 95% CIs were 5.44, 0.08, [5.27, 5.61] for normal urine samples and were 6.79, 0.13, [6.54, 7.03] for cancer urine samples. The p-values of two-sample Wilcoxon tests for these three methods were all < 0.001.