Supporting information for

Reversible colorimetric sensing of volatile analytes by wicking in close proximity to a photonic film

Timothy J. Palinski,[†#a] Bin Guan,[#b, c] Bronwyn H. Bradshaw-Hajek,[c] Michael A. Lienhard,[a] Craig Priest[*b, c, d] and Félix A. Miranda [a]

[a]Communications & Intelligent Systems Division, NASA Glenn Research Center, Cleveland, Ohio 44135, USA

[b]Future Industries Institute, University of South Australia, Mawson Lakes, SA 5095, Australia

[c]UniSA STEM, University of South Australia, Mawson Lakes, SA 5095, Australia

[d]Australian National Fabrication Facility – South Australia Node, University of South Australia, SA 5095, Australia

[†]Current Address: Honeywell International, Inc., Broomfield, Colorado 80021, USA.

[#] The authors contributed equally to this work.

* Corresponding authors

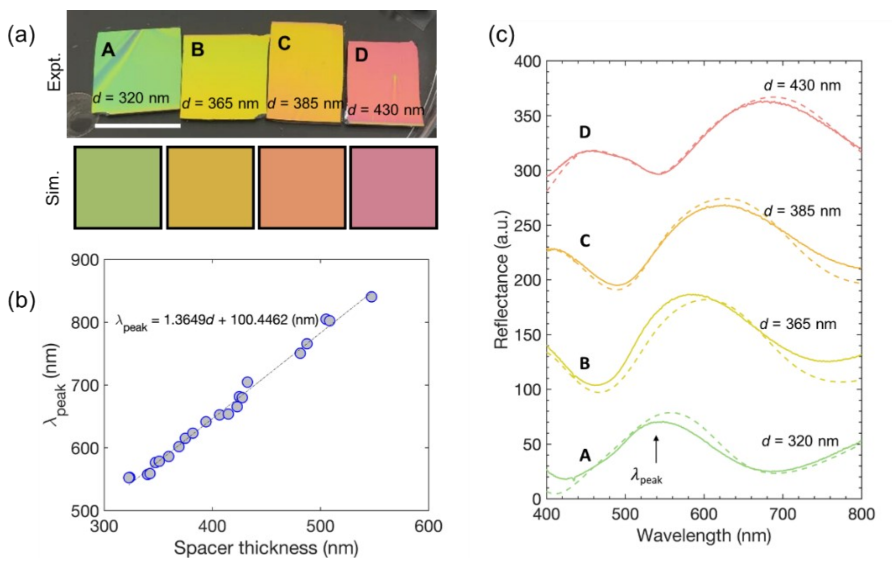

Fig. S1 Nanoisland MIM sensor design and characterization. (a) Color photographs of several calibration samples with different spacer thicknesses d (top); simulated color palette (bottom). Scale bar 1 cm. The CIE 1931 chromaticity calculations were produced using a free, open-source MATLAB code.[1] The color palette corresponds to spectra calculated via the TMM at cavity thicknesses (d) matching the experimental images above. (b) Calibration curve derived from ellipsometry-measured spacer thicknesses and first-order reflectance peak locations λ_{peak}. (c) Measured (solid) and transfer matrix method-simulated (dashed) reflectance spectra. Curves offset for clarity.

MATLAB code for color to thickness conversion

```
%Author: TJP
%Date: 2/17/2022
%File: Calc_best_RGB_fit.m
%Purpose: map test RGB vector to spacer thickness calibration dataset using
% 'best fit' match. Currently has programmed Cosine simililarity and
% Euclidean distance methods.
%
% Other matching methods can be found here/implemented later:
%
% A. Sanda-Mahama, et al.,
%    "Choice of distance metrics for RGB color image analysis,"
%     Soc. Imag. Sci. & Tech., 2016
%

function [mv, I, f] = Calc_best_RGB_fit(Cm, Dm, ls, lf, t_arr, fit_type)
    %inputs:
    %       Cm                      Calibration matrix
    %       Dm                      RGB data matrix
    %       t_arr     1 x n_images  array of time stamps
    %       ls        double        starting wavlength 'lambda start'
    %                                   for fitting algorirthm
    %       lf        double        final wavelength 'lambda final'
    %                                   for fitting algorithm
    %       fit type  string        "Cosine similarity" or
    %                                   "Euclidean distance"
    %                               Can specify other fits in if/else below
    %outputs:
    %       mv        1 x n_expt        array of min (or max) values
    %                                       depending on fit type
    %       I         1 x n_expt        array min (or) max indices
    %       f         n_pts x n_expt    array of fit data


    if fit_type == "Cosine similarity"
        for i=1:length(t_arr)
            test_RGB = Dm(i,:);
            cs(:,i) = Cosine_similarity(test_RGB, Cm);
        end
        f = cs;
    elseif fit_type == "Euclidean distance"
        p = 2; %p=2 --> Euclidean distance calc from Minkowsky formula
        for i=1:length(t_arr)
            test_RGB = Dm(i,:);
            mk(:,i) = Minkowsky(test_RGB, Cm, p);
        end
        f = mk;
    else
        disp('Error 1: no fit information!')
    end


    if fit_type == "Cosine similarity"
        for i=1:length(t_arr)
            %find local max cosine sim
            [mv(i), I(i)] = max(cs(ls:lf,i));
        end
    elseif fit_type == "Euclidean distance"
        for i=1:length(t_arr)
            %find local min Minkowsky (distance)
            [mv(i), I(i)] = min(mk(ls:lf,i));
        end
    else
        disp('Error 2: no fit information!')
    end

end
```

```
function mk = Minkowsky(v1,cal_matrix, p)
    %inputs:
    %       v1           1 x 3      test RGB vector
    %       cal_matrix = n x 3      matrix of RGB values
    %       p            int        Minkowsky factor
    %       p=1 --> city block distance; p=2 --> Euclidian distance
    %output:
    %       Minkowsky 'specra' btwn v1 and each vector in cal_martrix

    [n,m] = size(cal_matrix);

    for i=1:n

        v2 = cal_matrix(i,:);
        mk(i) = (abs((v1(1)-v2(1)))^p + abs((v1(2)-v2(2)))^p + ...
            abs((v1(3)-v2(3)))^p)^(1/p);

    end


end
```

```
function s = Cosine_similarity(v1,cal_matrix)
    %inputs:
    %       v1           1 x 3      test RGB vector
    %       cal_matrix = n x 3      matrix of RGB values
    %       p            int        Minkowsky factor
    %output:
    %       Cosine sim match btwn v1 and each vector in cal_martrix

    [n,m] = size(cal_matrix);

    for i=1:n

        v2 = cal_matrix(i,:);
        mag_v1 = norm(v1);
        mag_v2 = norm(v2);
        s(i) = dot(v1, v2)/(mag_v1*mag_v2);

    end

end
```

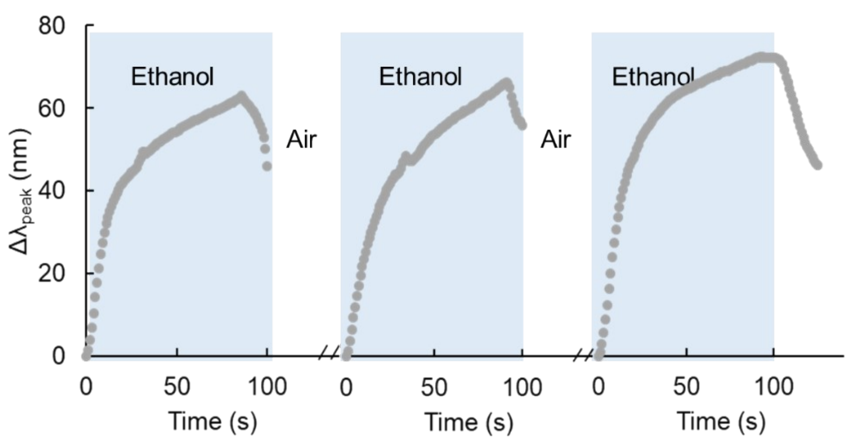

Fig. S2 The reversibility of the optofluidic sensor to ethanol exposure via the micropillar array. The sensor response represented by the reflectance peak shift over a few cycles of 100% ethanol and air exposure. The reversible sensing experiments were conducted with a smaller liquid reservoir to only sustain about 80 s of ethanol exposure. The decrease in peak shift in each circle indicates the exhaustion of ethanol in the reservoir. The air exposure time is larger than 30 min.

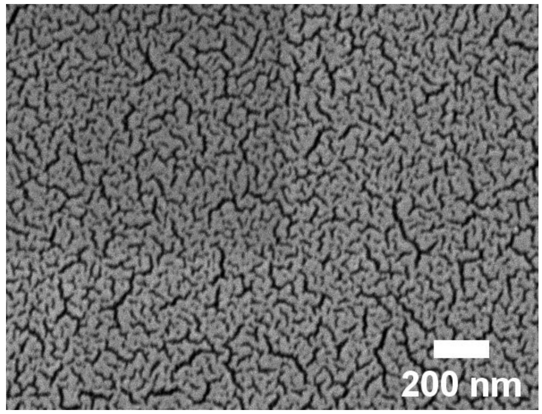

Fig. S3 The scanning electron micrograph of the porous nanoisland film of the fabricated MIM sensors. The bright areas are Au nanoislands, separated by the dark gaps in between. The nanoislands only cover ~ 73% of the area as calculated by the image processing software ImageJ.

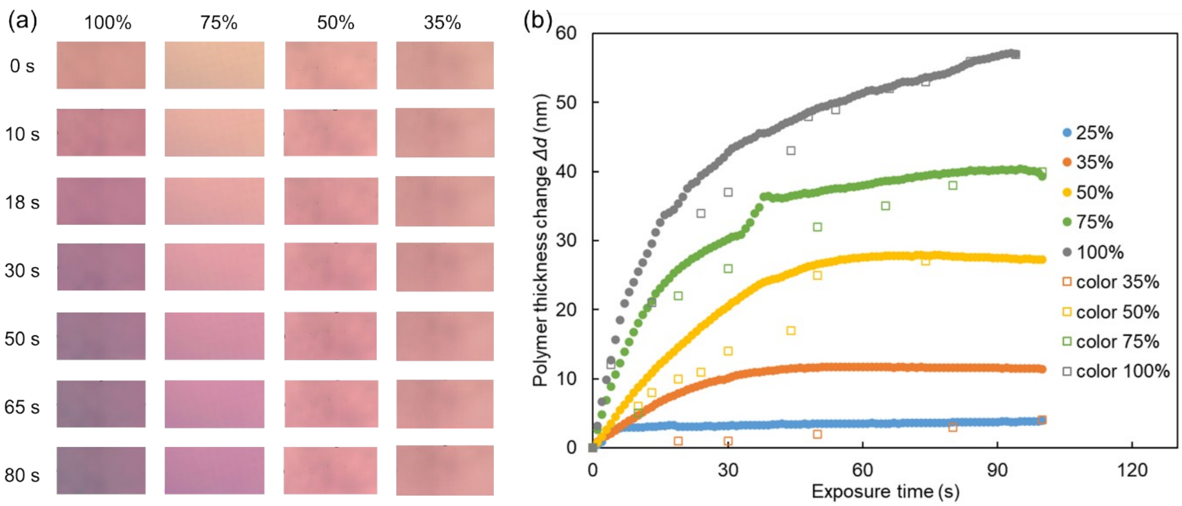

Fig. S4 (a) Color photographs of the MIM sensor as functions of ethanol concentration in weight percentage and time. (b) The sensor response to different ethanol concentrations (in weight percentage) by RGB color analysis and spectroscopic measurement. The increase of polymer thickness plotted over time when different concentrations of ethanol liquid were introduced into the pillar array.

References

1. M. Aldrich, pspectro: Photometric and Colorimetric Calculations, https://www.mathworks.com/matlabcentral/fileexchange/28185-pspectro-photometric-and-colorimetric-calculations, MATLAB Central File Exchange, (accessed February 28, 2020).