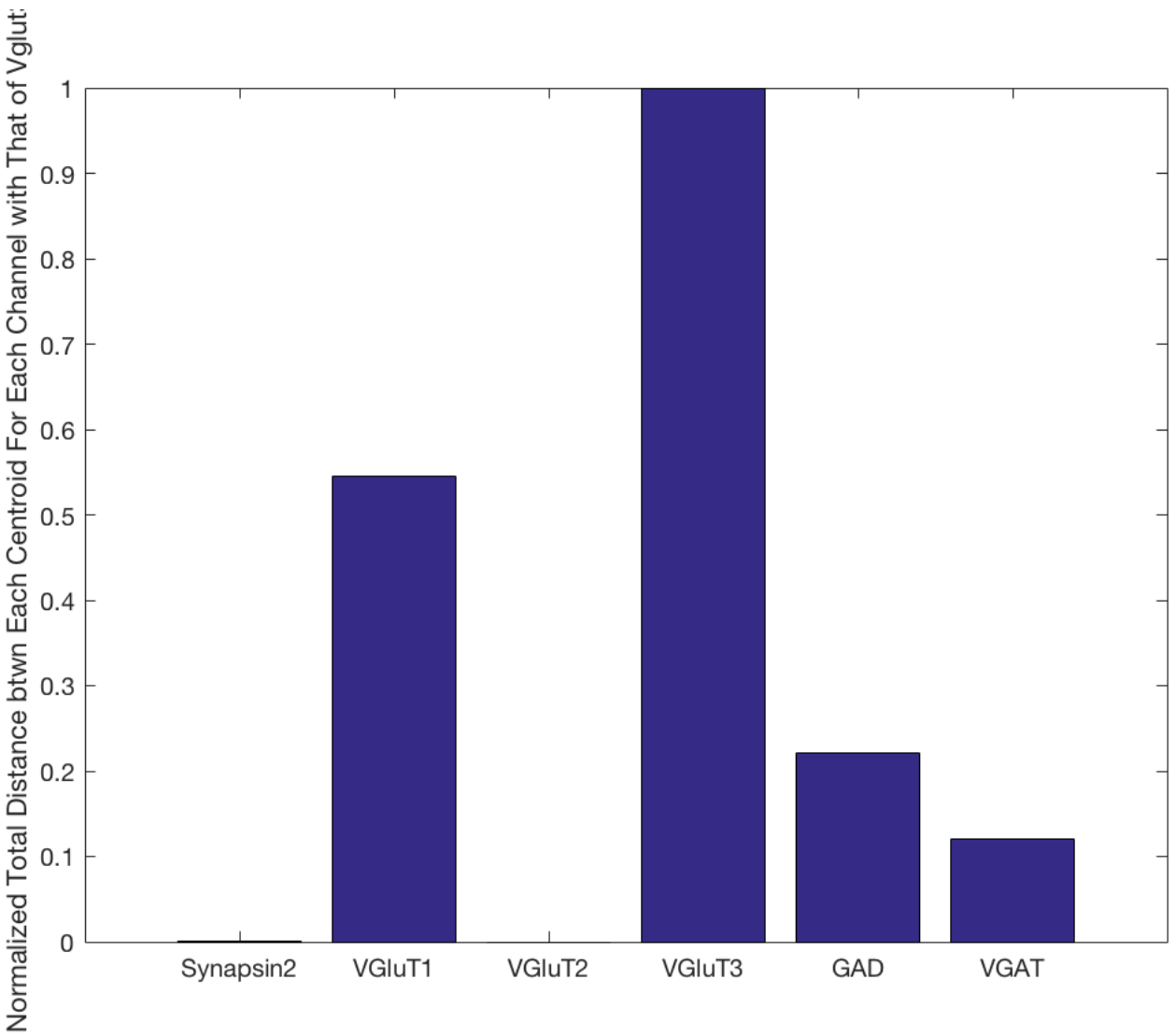
```
% perform keman clustering of the called hospots positions
Centers = {[], [], [], [], [], []};

for channel = 1 : 6

    for raw_image = 1 : 41

        temp = pointss{channel};

        if numel(temp) ~= 0
            [idx,C] = kmeans(temp, 20, 'Distance','cosine', 'Start', 'plus');
        end

        Centers{channel} = [Centers{channel}; C, raw_image * ones(20, 1)];

    end
end

%use the vglut1 channel  (an excitory synapse marker) as a baseline;
%compare it with the centroids from other channels and try to minimize the
% distance between the centroids

vglut1 = Centers{3};
Distance = [1 , 2 , 3, 4, 5, 6];
for channel = [1, 2, 3, 4, 5, 6]


    temp1 = Centers{channel};

    for raw_image = 1 : 41
        temp_distance = 0;
        temp = [];
        for temp1_center = 1 : 20
            for vglut_center = 1 : 20
                temp = [temp; (temp1(temp1_center + (raw_image - 1) * 20) - vglut1(vglut_cente
r + (raw_image - 1) * 20)).^2, vglut_center, temp1_center];
            end

        end


        [distance, pos] = min(temp(:,1));
        temp(pos, 1) = 1000;
        vlgut1_pos = temp(pos, 2);
        temp1_pos = temp(pos, 3);
        temp(find(temp(:,2) ==  vlgut1_pos), :) = [];
        temp(find(temp(:,3) == temp1_pos), :) = [];

        num_pos = 19;

        while (num_pos > 0)

            num_pos = num_pos - 1;
            [distance, pos] = min(temp(:,1));
            temp(pos, 1) = 1000;
```

```
            vlgut1_pos = temp(pos, 2);
            temp1_pos = temp(pos, 3);
            temp(find(temp(:,2) ==  vlgut1_pos), :) = [];
            temp(find(temp(:,3) == temp1_pos), :) = [];
            distance = distance1 + distance;

        end

        temp_distance = temp_distance + distance;

    end
    Distance(channel) = temp_distance;

end

bar(Distance / max(Distance));
set(gca, 'Xticklabel',{'Synapsin2', 'VGluT1', 'VGluT2', ...
    'VGluT3', 'GAD', 'VGAT' })
ylabel('Normalized Total Distance btwn Each Centroid For Each Channel with That of Vglut3')
```



Published with MATLAB® R2016a