
Bilinear Scan Conversion via sequential 'slicing' of 3d data

Scan converts spherical ultrasound data to cartesian with bilinear interpolation.

```
function scanConvert3DVolume_sliceMethod( obj )
%SCANCONVERT3DVOLUME_SLICEMETHOD Summary of this function goes here
%   Detailed explanation goes here
%   % 3d scan convert multistage from 2d scan convert
%   % define polar dimension arrays
    rk = linspace(obj.rmin,obj.rmax,size(obj.rawData,1));
%   thetak =
    asin((linspace(sin(obj.thetamin),sin(obj.thetamax),size(obj.rawData,2)))));
%   phik =
    asin(linspace(sin(obj.phimin),sin(obj.phimax),size(obj.rawData,3)));
    thetak = (linspace((obj.thetamin),
(obj.thetamax),size(obj.rawData,2))));
    phik = linspace((obj.phimin),(obj.phimax),size(obj.rawData,3));
    % define grids
    [r_2d_points,theta_2d_points] = ndgrid(rk,thetak);
    [x_2d_points,y_2d_points ] =
pol2cart(theta_2d_points,r_2d_points);
    minX = min(x_2d_points(:));
    maxX = max(x_2d_points(:));
    minY = min(y_2d_points(:));
    maxY = max(y_2d_points(:));
    obj.dr = abs(diff(rk));
    obj.dr = obj.dr(1);
    obj.dx = obj.dr*obj.cartScalingFactor;
    obj.dy = obj.dr*obj.cartScalingFactor;
    obj.dz = obj.dr*obj.cartScalingFactor;
    obj.x_range = minX:obj.dx:maxX;
    obj.y_range = minY:obj.dy:maxY;
    obj.z_range = minY:obj.dy:maxY;
    [x_2d_mat,y_2d_mat] = ndgrid(obj.x_range,obj.y_range);

    interVolume =
zeros(size(x_2d_mat,1),size(x_2d_mat,2),size(obj.rawData,3),size(obj.rawData,4));
    for currVolume = 1:size(obj.rawData,4)
        for phi = 1:size(obj.rawData,3)
            interVolume(:, :, phi, currVolume) =
griddata(x_2d_points,y_2d_points,obj.rawData(:, :, phi, currVolume),x_2d_mat,y_2d_ma
            %imagesc(squeeze(abs(interVolume(:, :, phi, currVolume))));
            %pause(.01);
        end
    end
    r_2_arr = linspace(obj.rmin,obj.rmax,size(interVolume,1));
    [r_2_2d_points,phi_2d_points] = ndgrid(r_2_arr,phik);
    [x_2_2d_points,y_2_2d_points ] =
pol2cart(phi_2d_points,r_2_2d_points);
    minX = min(x_2_2d_points(:));
```

```

    maxX = max(x_2_2d_points(:));
    minY = min(y_2_2d_points(:));
    maxY = max(y_2_2d_points(:));
    [x_2_2d_mat,y_2_2d_mat] =
    ndgrid(minX:obj.dx:maxX,minY:obj.dy:maxY);

    % interVolume(x',y',phi)
    % y' is the new scan line
    %
    for currVolume = 1:size(obj.rawData,4)
        for theta = 1:size(interVolume,2)
            finalVolume(:,theta,:,currVolume) =
            griddata(x_2_2d_points,y_2_2d_points,squeeze(interVolume(:,theta,:,currVolume)),x
                %imagesc(squeeze(abs(finalVolume(:,theta,:,currVolume)))));
                %pause(.1);
            end
        end
        obj.rawData_cart_slicemethod = finalVolume;
    end

```

Published with MATLAB® R2016a