```
function [joint_angles_1,joint_angles_2,joint_angles_3] = getFingerAngles(contact_pts,body_config,base_pts,rpy,body_consts,RRR_phi)
addpath('/Users/guoyongxin/Desktop/Assignment_Academics/Senior_Second semester/MEC529/Myfunctions');

% body_consts
l1_1 = body_consts(1);
l2_1 = body_consts(2);
l1_2 = body_consts(3);
l2_2 = body_consts(4);
l3_2 = body_consts(5);
l1_3 = body_consts(6);
l2_3 = body_consts(7);
l3_3 = body_consts(8);
% contact pts
contact_pt_1 = contact_pts(:,1);
contact_pt_2 = contact_pts(:,2);
contact_pt_3 = contact_pts(:,3);
% body_config
T_OToP = body_config;
% rpy
rpy_1 = rpy(:,1);
rpy_2 = rpy(:,2);
rpy_3 = rpy(:,3);
% get rotation matrix
R_1 = RPY_to_Rot(rpy_1);
R_2 = RPY_to_Rot(rpy_2);
R_3 = RPY_to_Rot(rpy_3);
% get position vector;
P_1 = base_pts(:,1);
P_2 = base_pts(:,2);
P_3 = base_pts(:,3);
% Assign transformation matrix T_FiToP from finger frame {Fi} to {P}
z = [0 0 0];
T_F1ToP = [R_1,P_1;z,1];
T_F2ToP = [R_2,P_2;z,1];
T_F3ToP = [R_3,P_3;z,1];
% get inverse of T_FiToP
T_F1ToP_inv = getTransInv(T_F1ToP);
T_F2ToP_inv = getTransInv(T_F2ToP);
T_F3ToP_inv = getTransInv(T_F3ToP);
% Compute contact point expressed in frame {Fi}.
contact_pt_1_F1 = T_F1ToP_inv * T_OToP * [contact_pt_1;1];
contact_pt_2_F2 = T_F2ToP_inv * T_OToP * [contact_pt_2;1];
contact_pt_3_F3 = T_F3ToP_inv * T_OToP * [contact_pt_3;1];


% use RR and RRR IK algorithms
joint_angles_1 = RR_inverse_2D([l1_1;l2_1],contact_pt_1_F1(2:3));
joint_angles_2 = RRR_inverse_2D([l1_2;l2_2;l3_2],[contact_pt_2_F2(2:3);RRR_phi(1)]);
joint_angles_3 = RRR_inverse_2D([l1_3;l2_3;l3_3],[contact_pt_3_F3(2:3);RRR_phi(2)]);

% offset the joint angle since zero reference line is different, as
% illustrated in the midterm paper. The offset value is -pi/2.
if  isstring(joint_angles_1) == 1
    % do nothing, point not accessible.
else
    joint_angles_1(1,:) = joint_angles_1(1,:) - pi/2;
end



if  isstring(joint_angles_2) == 1
    % do nothing, point not accessible.
else
    joint_angles_2(1,:) = joint_angles_2(1,:) - pi/2;
end



if  isstring(joint_angles_3) == 1
    % do nothing, point not accessible.
else
    joint_angles_3(1,:) = joint_angles_3(1,:) - pi/2;
end


end
```

```
Not enough input arguments.

Error in getFingerAngles (line 5)
l1_1 = body_consts(1);
```

Published with MATLAB® R2018b