

```

                                extract_3D_features
function [BX C X Y Z S region_matrix N S_mat prob_mat] = extract_3D_features(I)

BW = rgb2gray(I);

BW = double(BW)/255;

[region_matrix N S_mat prob_mat] = identify_fine_fast(I);

im_size = size(I);
num_rows = im_size(1);
num_cols = im_size(2);
row_length = floor(num_rows/N);
col_length = floor(num_cols/N);

depth = log2((num_rows + num_cols)/2);

C1 = double(I(:,:,1))/255;
C2 = double(I(:,:,2))/255;
C3 = double(I(:,:,3))/255;

S = 9*ones(1,num_rows*num_cols);

P = depth*ones(num_rows,num_cols);

for i = 1 : N
    for j = 1 : N
        P((i-1)*row_length + 1 : i*row_length, (j-1)*col_length + 1 : j*col_length) =
(depth - depth*prob_mat(i,j));
    endfor
endfor

for i = 1 : num_rows
    X((i-1)*num_cols + 1 : i*num_cols) = 1 : num_cols;
    Y((i-1)*num_cols + 1 : i*num_cols) = double(P(i,:));
    Z((i-1)*num_cols + 1 : i*num_cols) = (num_rows - i + 1)*ones(1,num_cols);
    B((i-1)*num_cols + 1 : i*num_cols,1) = BW(i,:);
    B((i-1)*num_cols + 1 : i*num_cols,2) = BW(i,:);
    B((i-1)*num_cols + 1 : i*num_cols,3) = BW(i,:);
    C((i-1)*num_cols + 1 : i*num_cols,1) = double(C1(i,:));

```

```
                                extract_3D_features
C((i-1)*num_cols + 1 : i*num_cols,2) = double(C2(i,:));
C((i-1)*num_cols + 1 : i*num_cols,3) = double(C3(i,:));

endfor

%this removes regions that are almost certainly part of the background of the image
temp_index = find(Y < .95*depth);

X = X(temp_index);
Y = Y(temp_index);
Z = Z(temp_index);

BX(:,1) = B(temp_index,1);
BX(:,2) = B(temp_index,2);
BX(:,3) = B(temp_index,3);

C = C(temp_index);

S = S(temp_index);

endfunction
```