

```

                                maximize_std_dev_fast
function [N_0, s_0, avg_ent] = maximize_std_dev_fast(input_im)
im_size = size(input_im);
alpha = 2.5;
num_rows = im_size(1);
num_cols = im_size(2);
N_0 = ceil(log2((num_rows + num_cols)/2));
N_1 = ceil((1 - alpha)*N_0);
[X Y] = test_measures_approx(input_im,N_0);
avg_ent = X;
s_0 = Y;
[X Y] = test_measures_approx(input_im,N_1);
s_1 = Y;
if(s_1 > s_0 )
    while(s_1 > s_0)
        N_0 = N_1;
        N_1 = ceil((1 - alpha)*N_0);
        [X Y] = test_measures_approx(input_im,N_0);
        avg_ent = X;
        s_0 = Y;
        [X Y] = test_measures_approx(input_im,N_1);
        s_1 = Y;
    endwhile
else
    N_1 = ceil((1 + alpha)*N_0);

```

```

                                maximize_std_dev_fast
[X Y] = test_measures_approx(input_im,N_0);
s_0 = Y;
[X Y] = test_measures_approx(input_im,N_1);
s_1 = Y;
while(s_1 >= s_0)
    N_0 = N_1;
    N_1 = ceil((1 + alpha)*N_0);
    [X Y] = test_measures_approx(input_im,N_0);
    avg_ent = X;
    s_0 = Y;
    [X Y] = test_measures_approx(input_im,N_1);
    s_1 = Y;
endwhile
endif
endfunction

```