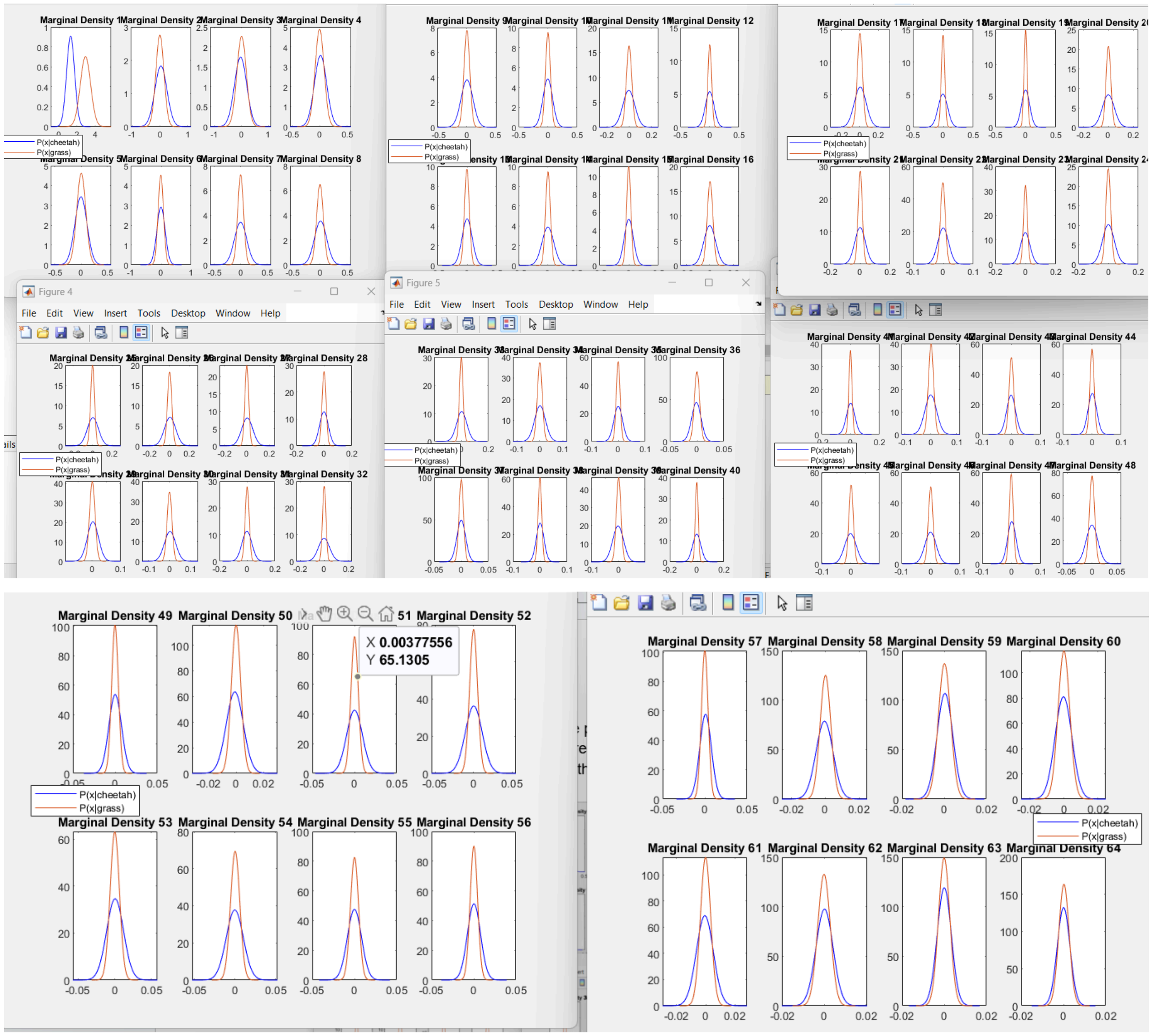
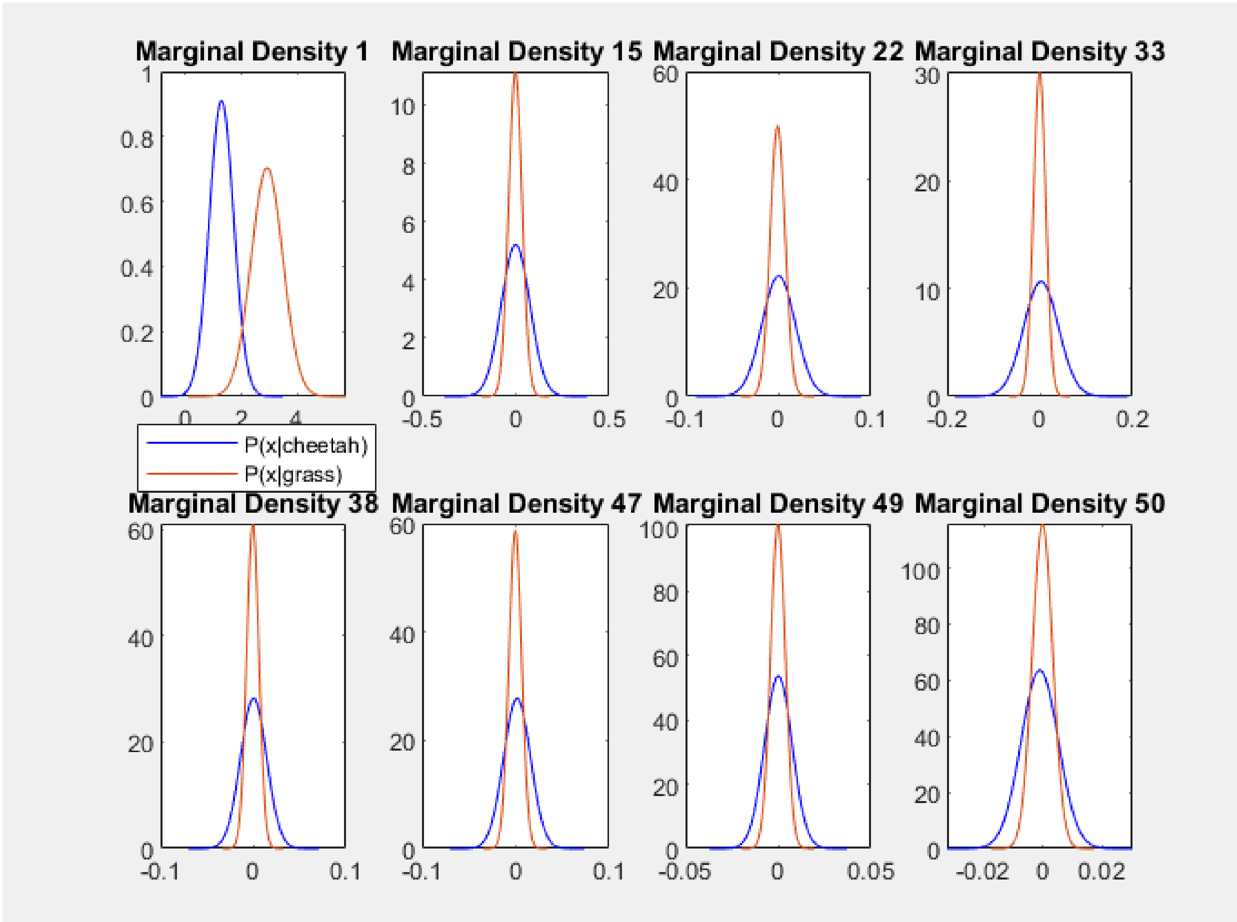
a) The maximum likelihood estimates of the prior probabilities are P(cheetah) = 0.1919 and P(grass) = 0.8081. These are the same results as HW1 and this means that the maximum likelihood priors are based on the proportion of the samples of the individual class with the total number of samples.

b)

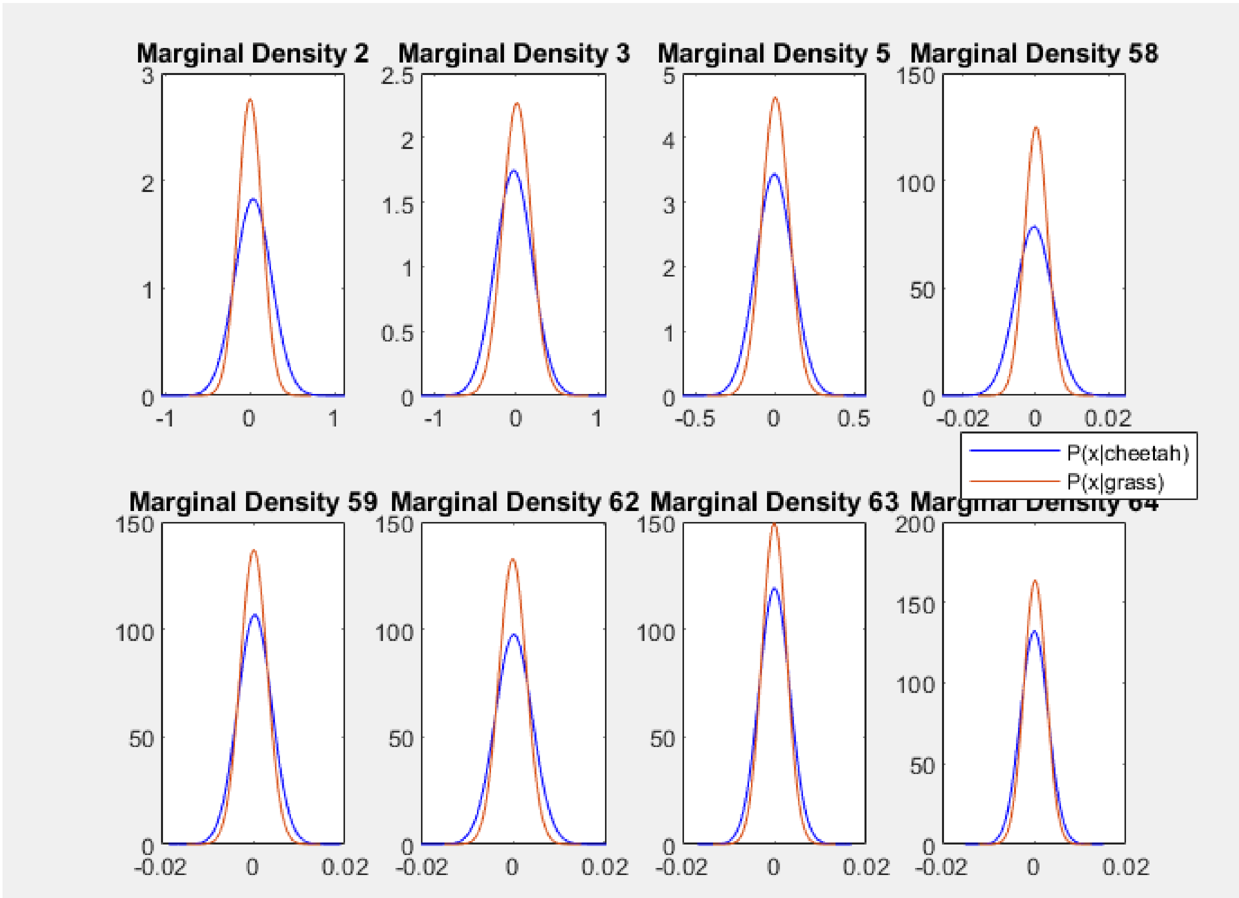


The best 8 features that I selected from visual inspection were Marginal Densities 1,15, 22, 33, 38, 47, 49, and 50.

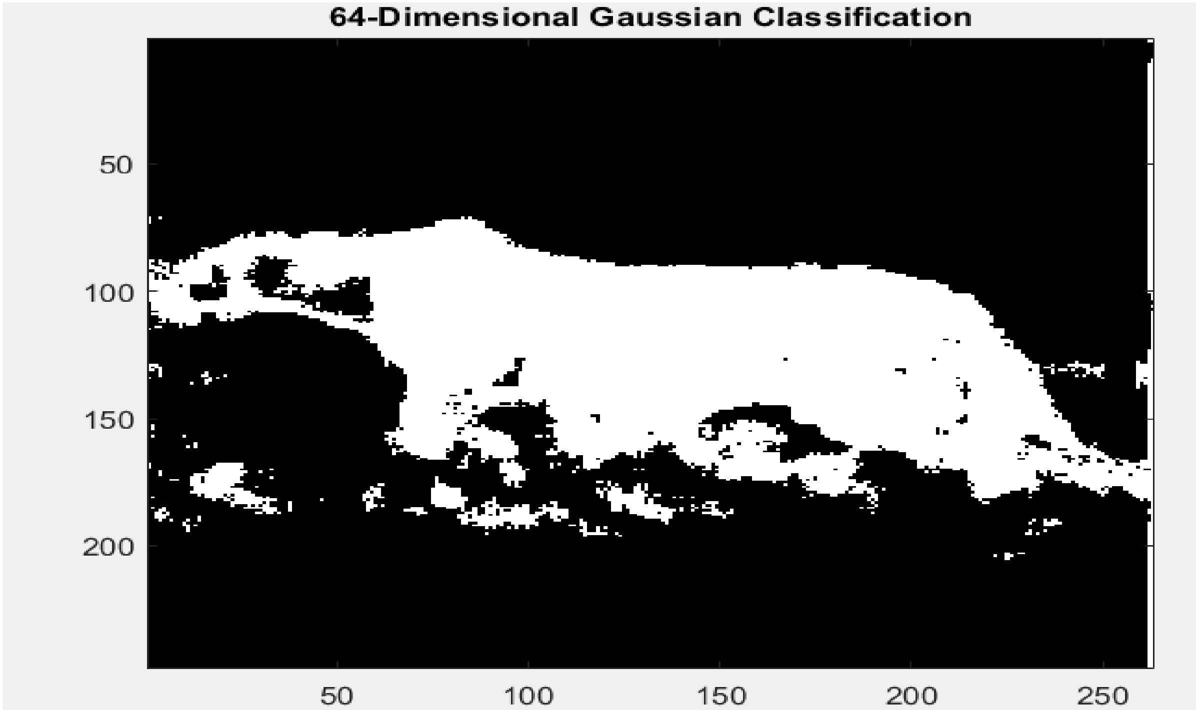


The worst 8 features that I selected were Marginal Densities 2, 3, 5, 58, 59, 62, 63, and

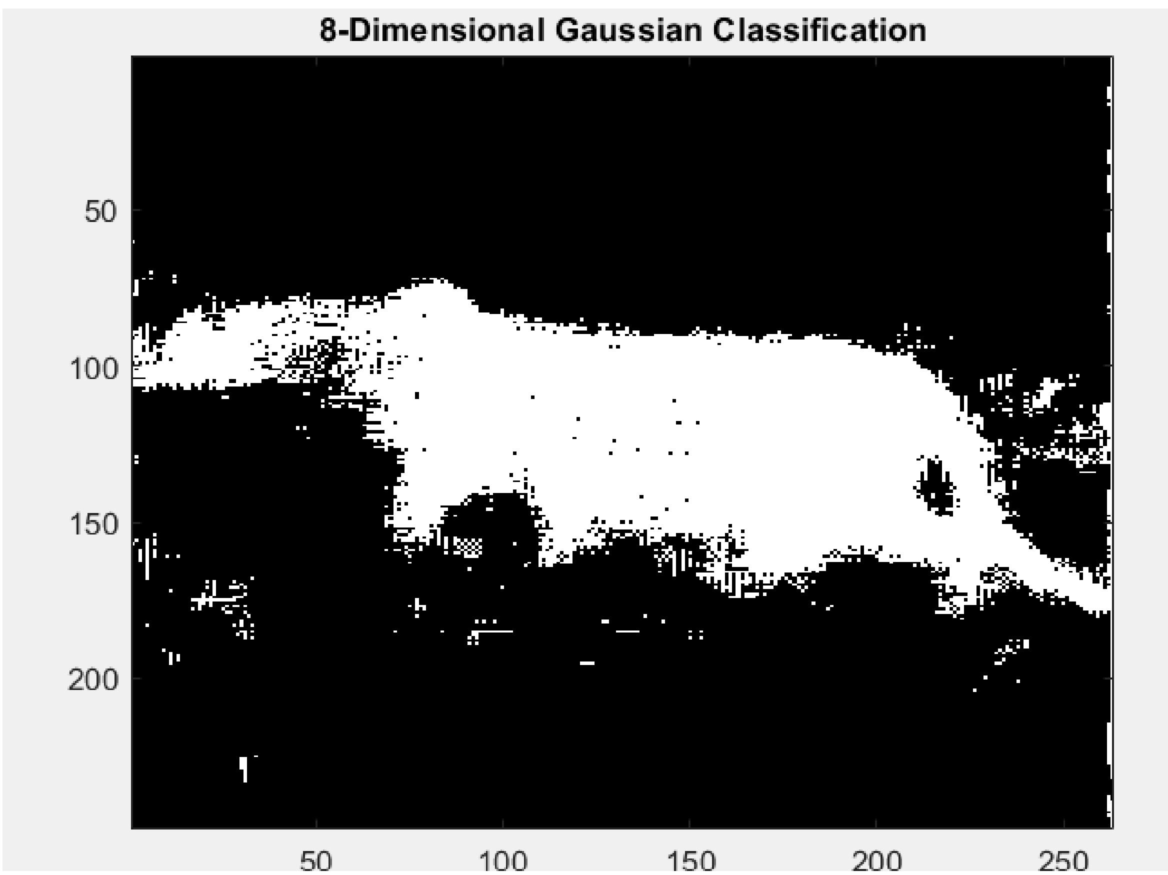
64.



c) The Bayesian Decision Rule is P(x|cheetah)*P(cheetah)* > *P(x|grass)*P(grass). If this is true, then the component of the picture will be marked as the cheetah. Otherwise, it will be marked as the grass/background. The image I generate from classifying the 64 class-conditional marginal densities is:



The error rate of this image is 8.01%. When I classify using the best 8 features, the image I get is:



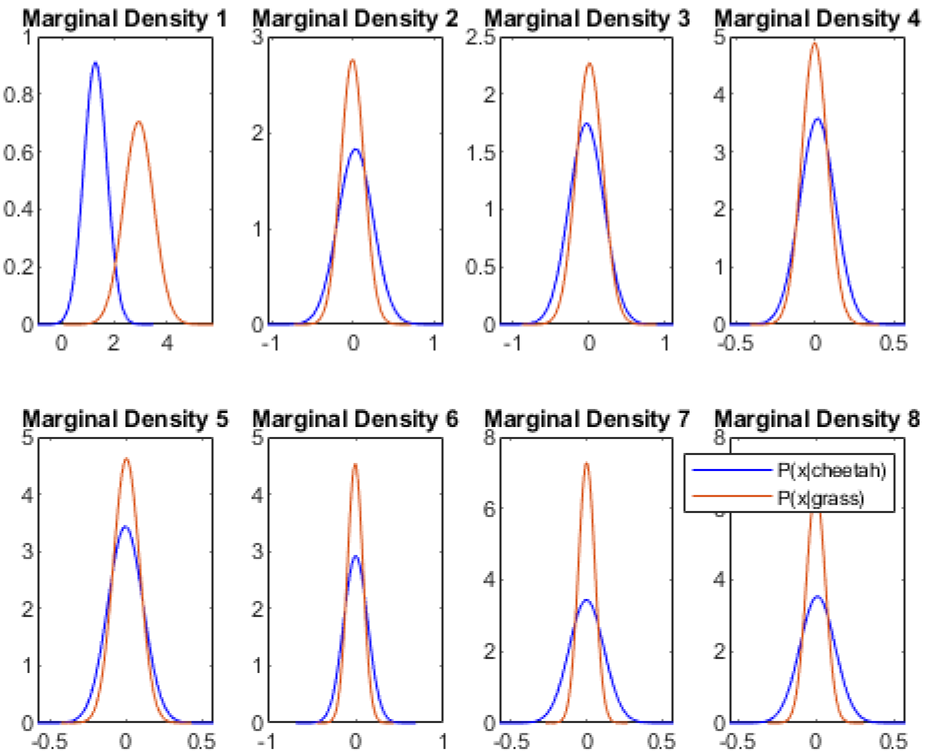
The error rate of this image is 6.03%. This image will do a better job of classifying as it is associated with the best 8 features and has less features that will come in the way of proper classification.

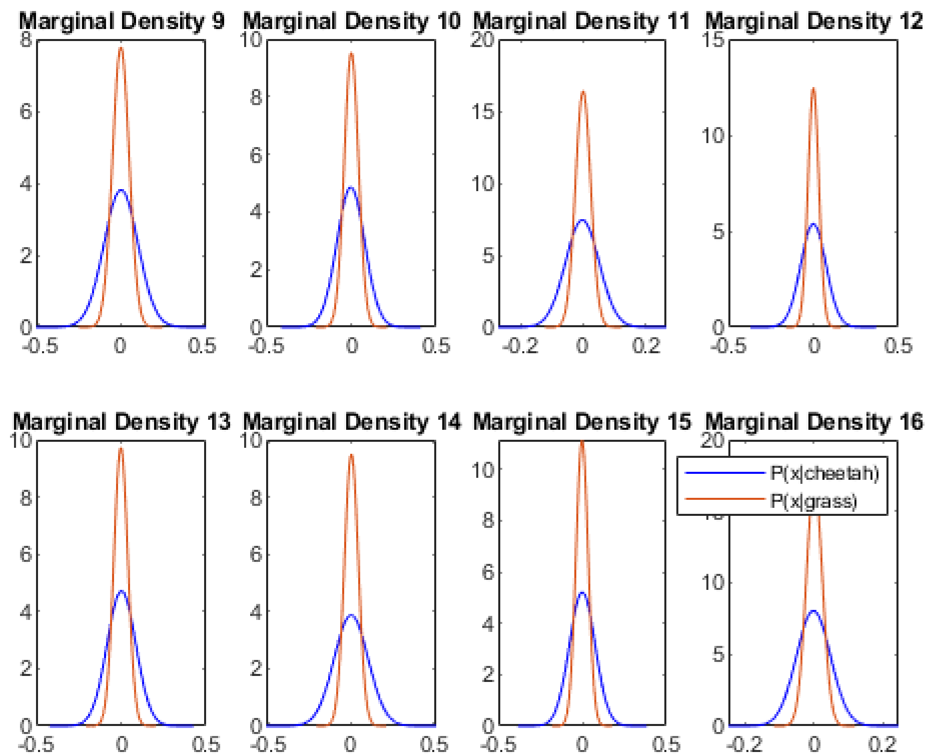
```
%a)
m = load('TrainingSamplesDCT_8_new.mat');
fg = m.TrainsampleDCT_FG;
bg = m.TrainsampleDCT_BG;
fg_rows = size(fg, 1);
bg_rows = size(bg, 1);
prior_cheetah = fg_rows / (fg_rows + bg_rows);
prior_grass = bg_rows / (fg_rows + bg_rows);
fprintf('P(Cheetah) = %.4f\n', prior_cheetah);
fprintf('P(Grass) = %.4f\n', prior_grass);
disp("The priors are the same as before.")

%b)
mean_fg = sum(fg) / fg_rows;
mean_bg = sum(bg) / bg_rows;
std_fg = std(fg);
std_bg = std(bg);

x_fg = zeros(64, 61);
y_fg = zeros(64, 61);
x_bg = zeros(64, 61);
y_bg = zeros(64, 61);

for feature_idx = 1:64
    x_fg(feature_idx, :) = linspace(mean_fg(feature_idx) - 5 *
std_fg(feature_idx), mean_fg(feature_idx) + 5 * std_fg(feature_idx), 61);
    y_fg(feature_idx, :) = normpdf(x_fg(feature_idx, :),
mean_fg(feature_idx), std_fg(feature_idx));
    x_bg(feature_idx, :) = linspace(mean_bg(feature_idx) - 5 *
std_bg(feature_idx), mean_bg(feature_idx) + 5 * std_bg(feature_idx), 61);
    y_bg(feature_idx, :) = normpdf(x_bg(feature_idx, :),
mean_bg(feature_idx), std_bg(feature_idx));
end

for fig_idx = 1:8
    figure;
    for subplot_idx = 1:8
        feature_num = (fig_idx - 1) * 8 + subplot_idx;
        subplot(2, 4, subplot_idx);
        plot(x_fg(feature_num, :), y_fg(feature_num, :), '-b',
x_bg(feature_num, :), y_bg(feature_num, :));
        title(['Marginal Density ', num2str(feature_num)]);
    end
    legend('P(x|cheetah)', 'P(x|grass)')
end

best_features = [1, 15, 22, 33, 38, 47, 49, 50];
worst_features = [2, 3, 5, 58, 59, 62, 63, 64];

figure;
for idx = 1:8
    subplot(2, 4, idx);
```

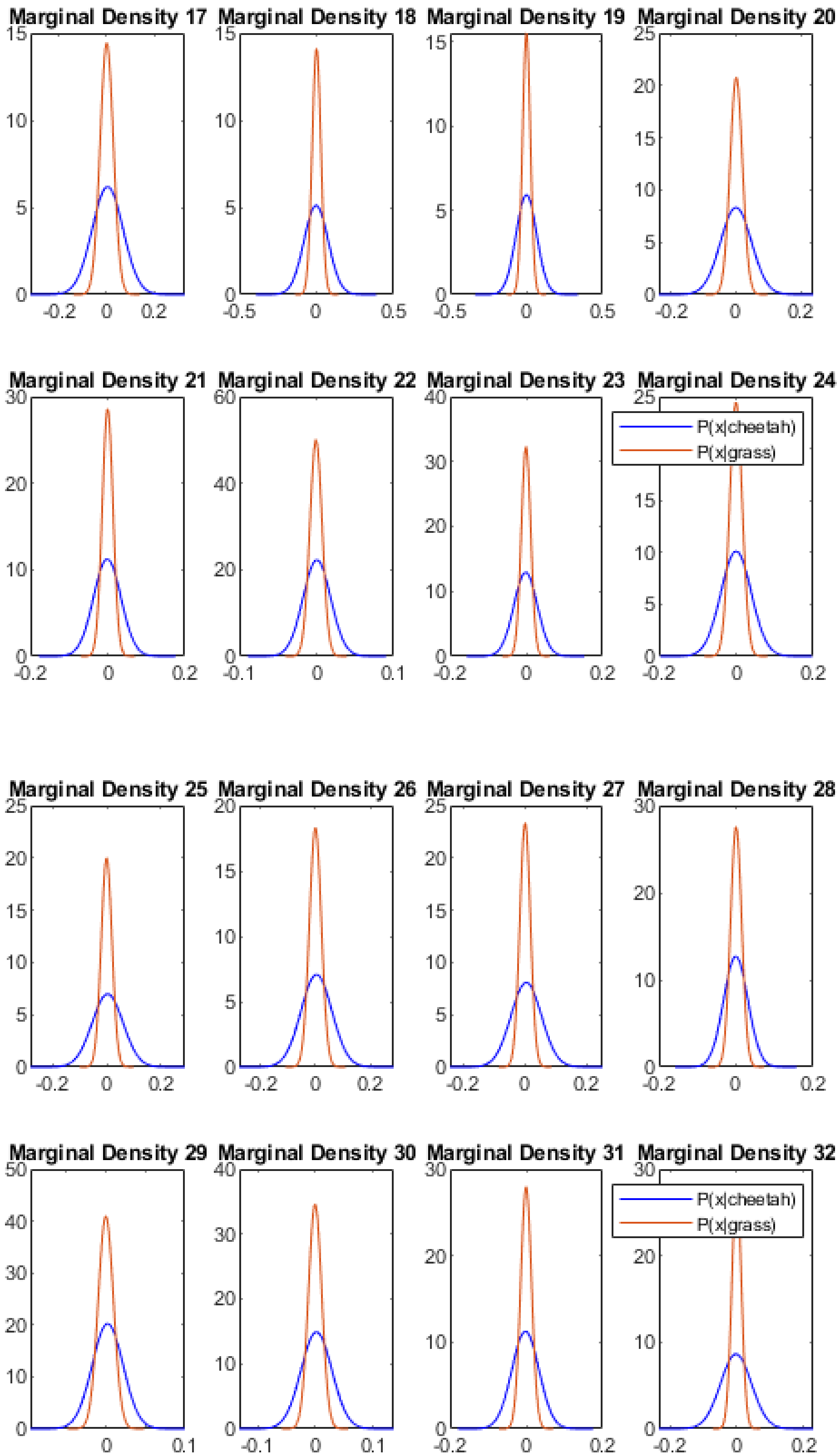
```
    plot(x_fg(best_features(idx), :), y_fg(best_features(idx), :), '-b',
x_bg(best_features(idx), :), y_bg(best_features(idx), :));
    title(['Marginal Density ', num2str(best_features(idx))]);
end
legend('P(x|cheetah)', 'P(x|grass)');

figure;
for idx = 1:8
    subplot(2, 4, idx);
    plot(x_fg(worst_features(idx), :), y_fg(worst_features(idx), :), '-b',
x_bg(worst_features(idx), :), y_bg(worst_features(idx), :));
    title(['Marginal Density ', num2str(worst_features(idx))]);
end
legend('P(x|cheetah)', 'P(x|grass)');

%c)
zigzag_indices = load('Zig-Zag Pattern.txt') + 1;
cheetah_image = im2double(imread('cheetah.bmp'));
[image_rows, image_cols] = size(cheetah_image);
small_constant = 1e-5;

bg_mean_matrix = repmat(mean_bg, bg_rows, 1);
fg_mean_matrix = repmat(mean_fg, fg_rows, 1);
covariance_bg = (bg - bg_mean_matrix)' * (bg - bg_mean_matrix) / bg_rows +
small_constant * eye(64);
covariance_fg = (fg - fg_mean_matrix)' * (fg - fg_mean_matrix) / fg_rows +
small_constant * eye(64);

result_64D = zeros(image_rows - 7, image_cols - 7);
result_8D = zeros(image_rows - 7, image_cols - 7);

for row = 1:(image_rows - 7)
    for col = 1:(image_cols - 7)
        dct_block = dct2(cheetah_image(row:row+7, col:col+7));

        feature_vector_64 = zeros(1, 64);
        for idx = 1:64
            [x_pos, y_pos] = find(zigzag_indices == idx);
            feature_vector_64(idx) = dct_block(x_pos, y_pos);
        end

        prob_bg = mvnpdf(feature_vector_64, mean_bg, covariance_bg) *
prior_grass;
        prob_fg = mvnpdf(feature_vector_64, mean_fg, covariance_fg) *
prior_cheetah;
        result_64D(row, col) = prob_fg > prob_bg;
    end
end

% Display image for 64-dimensional features
figure;
imagesc(result_64D);
title('64-Dimensional Gaussian Classification');
colormap gray(255);
```

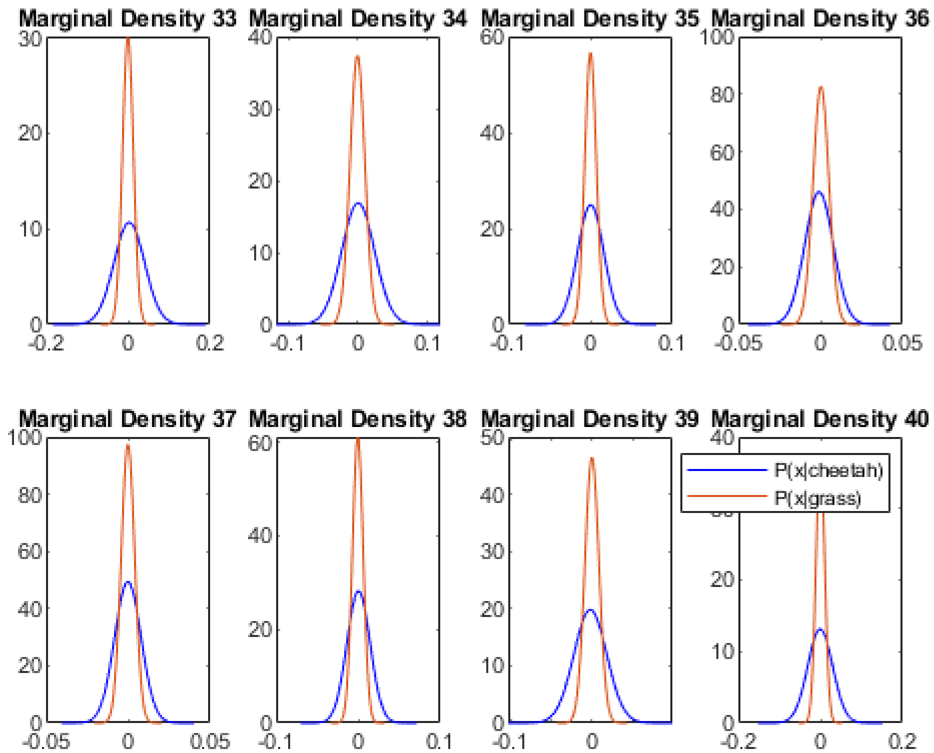
```
for row = 1:(image_rows - 7)
    for col = 1:(image_cols - 7)
        dct_block = dct2(cheetah_image(row:row+7, col:col+7));

        feature_vector_8 = zeros(1, 8);
        for idx = 1:8
            [x_pos, y_pos] = find(zigzag_indices == best_features(idx));
            feature_vector_8(idx) = dct_block(x_pos, y_pos);
        end

        prob_bg = mvnpdf(feature_vector_8, mean_bg(best_features),
covariance_bg(best_features, best_features)) * prior_grass;
        prob_fg = mvnpdf(feature_vector_8, mean_fg(best_features),
covariance_fg(best_features, best_features)) * prior_cheetah;
        result_8D(row, col) = prob_fg > prob_bg;
    end
end

% Display image for 8-dimensional features
figure;
imagesc(result_8D);
title('8-Dimensional Gaussian Classification');
colormap gray(255);

true_mask = im2double(imread('cheetah_mask.bmp'));
error_64D = sum(sum(abs(true_mask(1:image_rows-7, 1:image_cols-7) -
result_64D)));
error_8D = sum(sum(abs(true_mask(1:image_rows-7, 1:image_cols-7) -
result_8D)));
error_rate_64D = error_64D / ((image_rows - 7) * (image_cols - 7));
error_rate_8D = error_8D / ((image_rows - 7) * (image_cols - 7));

fprintf('Error Rate (64D) = %.4f\n', error_rate_64D);
fprintf('Error Rate (8D) = %.4f\n', error_rate_8D);
```

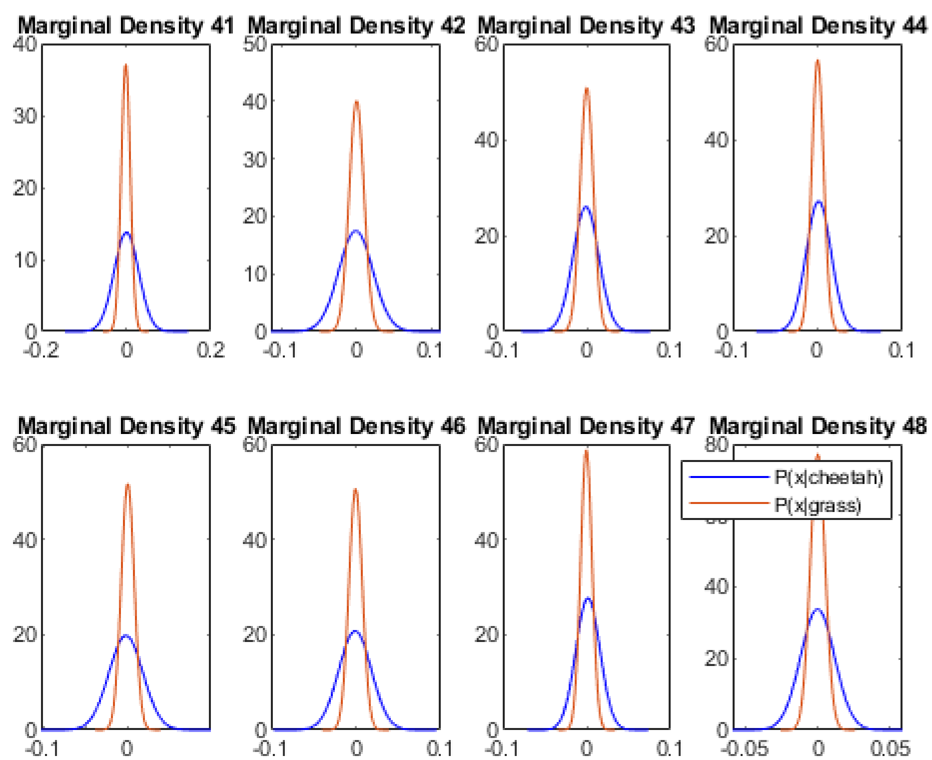
P(Cheetah) = 0.1919
P(Grass) = 0.8081
The priors are the same as before.
Error Rate (64D) = 0.0801
Error Rate (8D) = 0.0603

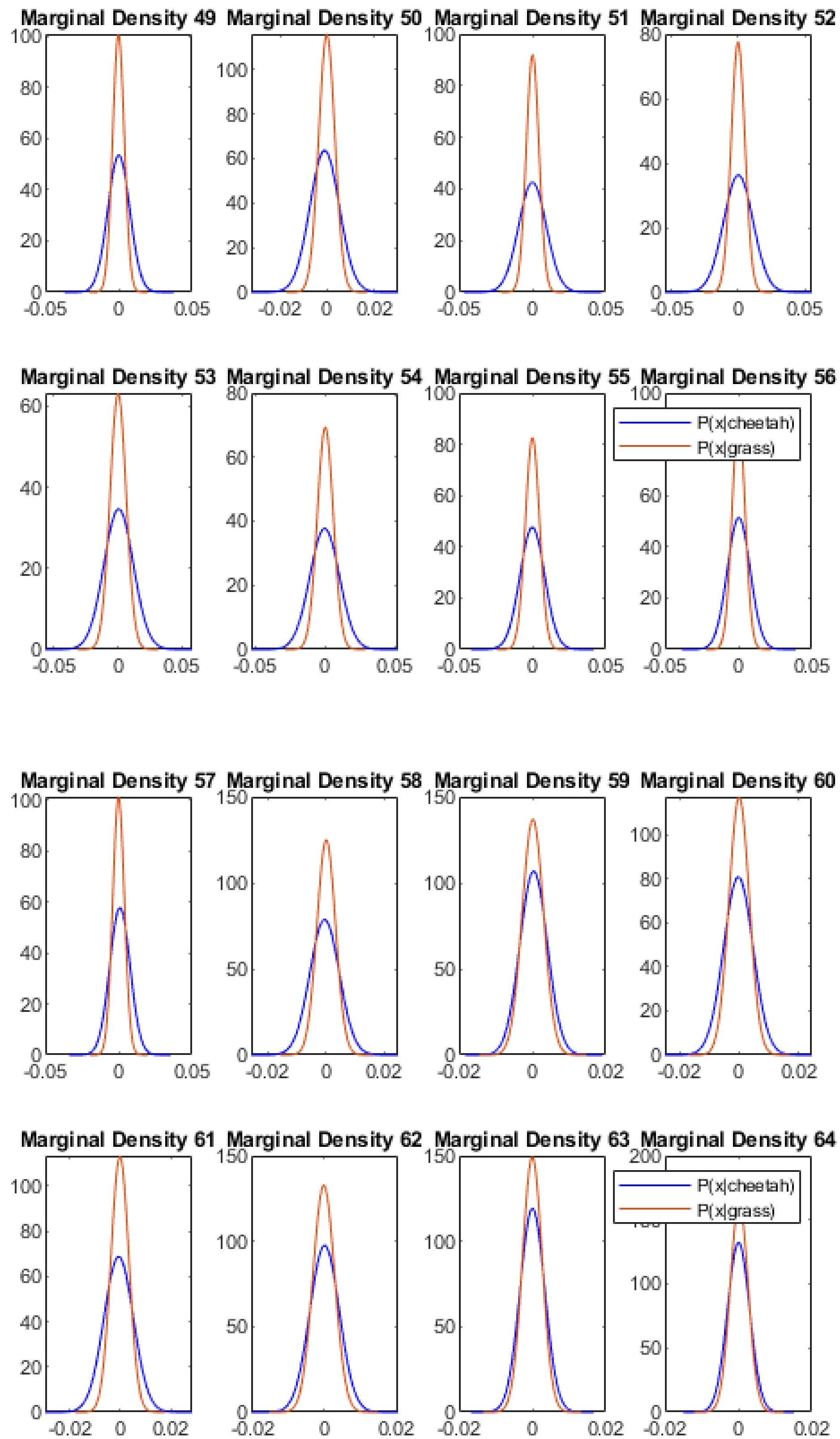

Marginal Density 1
Marginal Density 2
Marginal Density 3
Marginal Density 4
Marginal Density 5
Marginal Density 6
Marginal Density 7
Marginal Density 8
P(x|cheetah)
P(x|grass)

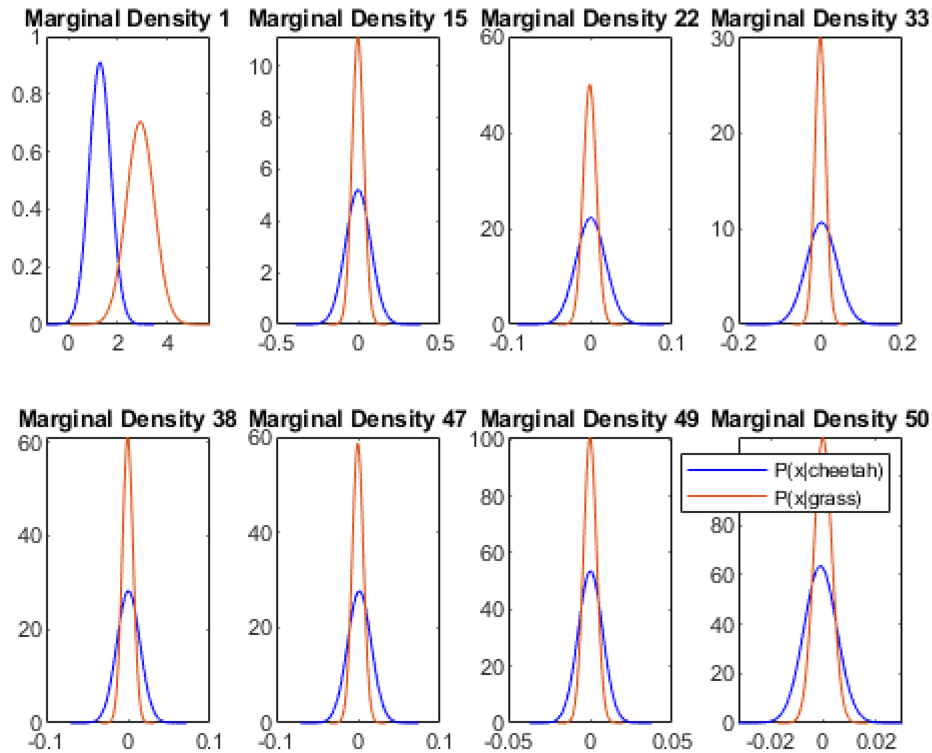

Marginal Density 9
Marginal Density 10
Marginal Density 11
Marginal Density 12
Marginal Density 13
Marginal Density 14
Marginal Density 15
Marginal Density 16
P(x|cheetah)
P(x|grass)

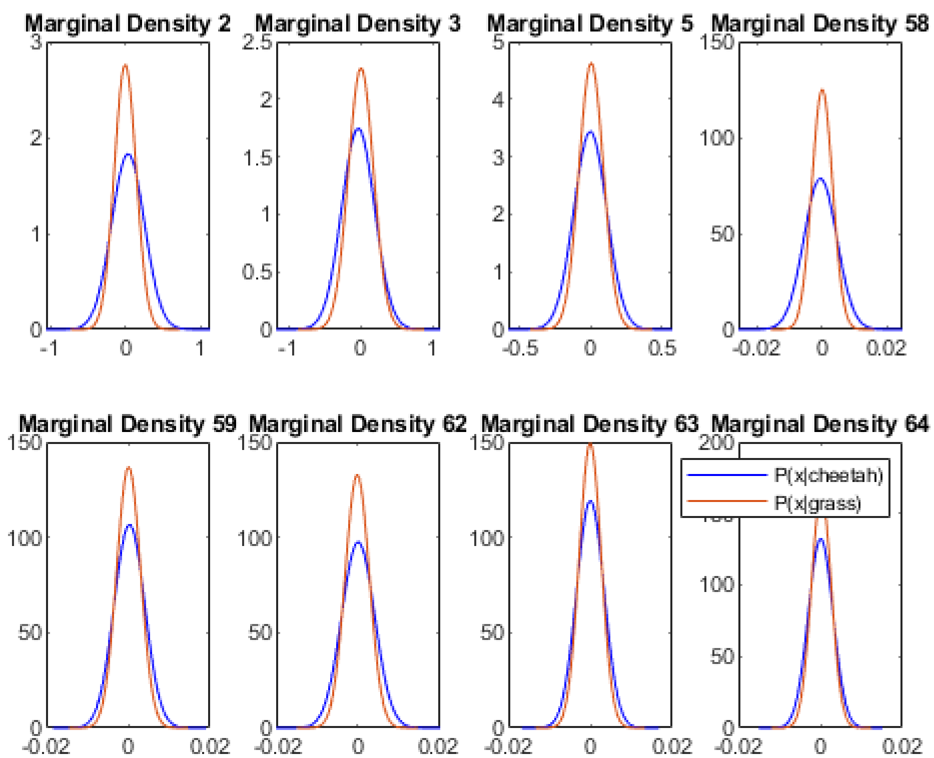

Marginal Density 17
Marginal Density 18
Marginal Density 19
Marginal Density 20
Marginal Density 21
Marginal Density 22
Marginal Density 23
Marginal Density 24
P(x|cheetah)
P(x|grass)
Marginal Density 25
Marginal Density 26
Marginal Density 27
Marginal Density 28
Marginal Density 29
Marginal Density 30
Marginal Density 31
Marginal Density 32
P(x|cheetah)
P(x|grass)

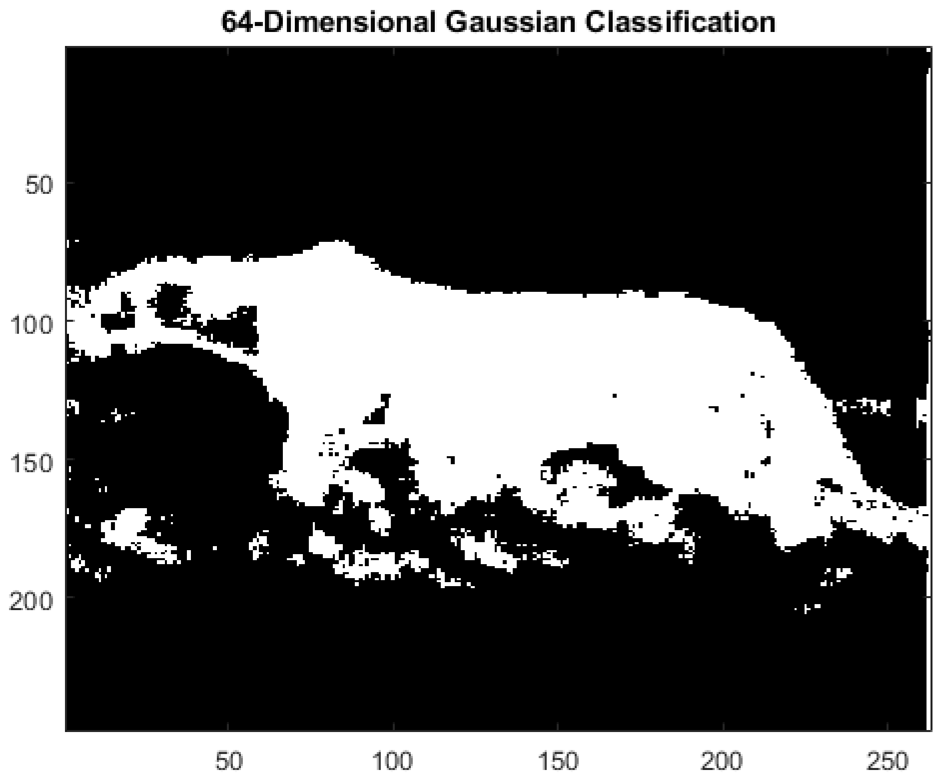

Marginal Density 33
Marginal Density 34
Marginal Density 35
Marginal Density 36
Marginal Density 37
Marginal Density 38
Marginal Density 39
Marginal Density 40
P(x|cheetah)
P(x|grass)

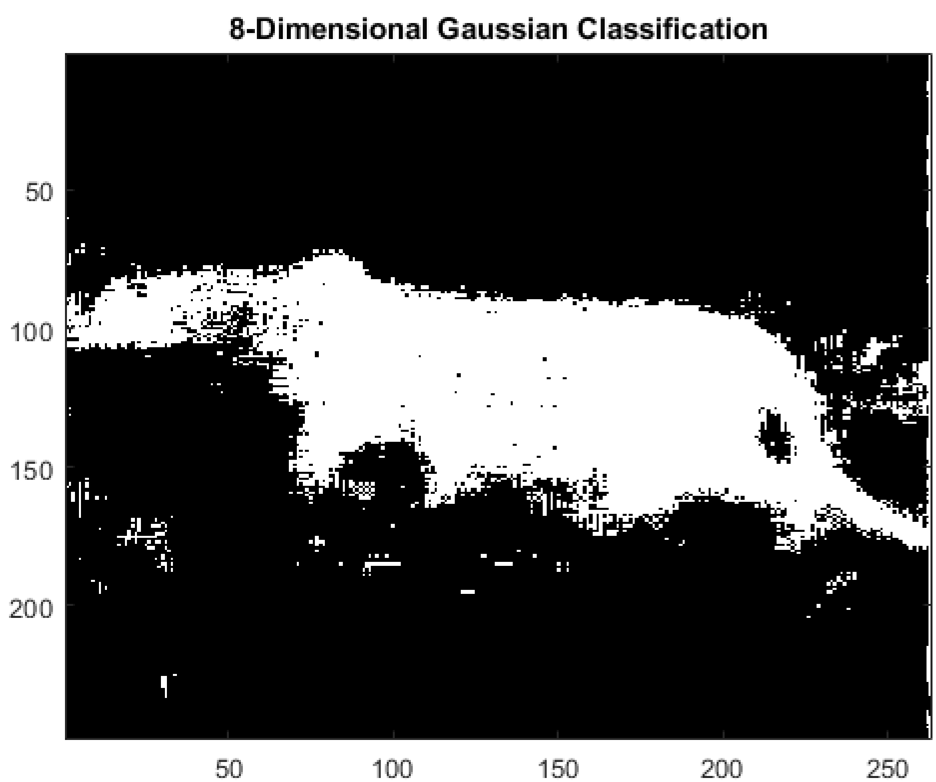

Marginal Density 41
Marginal Density 42
Marginal Density 43
Marginal Density 44
Marginal Density 45
Marginal Density 46
Marginal Density 47
Marginal Density 48
P(x|cheetah)
P(x|grass)


Marginal Density 49
Marginal Density 50
Marginal Density 51
Marginal Density 52
Marginal Density 53
Marginal Density 54
Marginal Density 55
Marginal Density 56
P(x|cheetah)
P(x|grass)
Marginal Density 57
Marginal Density 58
Marginal Density 59
Marginal Density 60
Marginal Density 61
Marginal Density 62
Marginal Density 63
Marginal Density 64
P(x|cheetah)
P(x|grass)


Marginal Density 1
Marginal Density 15
Marginal Density 22
Marginal Density 33
Marginal Density 38
Marginal Density 47
Marginal Density 49
Marginal Density 50
P(x|cheetah)
P(x|grass)


Marginal Density 2
Marginal Density 3
Marginal Density 5
Marginal Density 58
Marginal Density 59
Marginal Density 62
Marginal Density 63
Marginal Density 64
P(x|cheetah)
P(x|grass)


64-Dimensional Gaussian Classification
50
100
150
200
50
100
150
200
250


8-Dimensional Gaussian Classification
50
100
150
200
50
100
150
200
250

Published with MATLAB® R2024b