Detection of Uncovered Background and Moving Pixels

Jignesh Panchal, Member, IEEE
Department of Electrical & Computer Engineering, Mississippi State University

***Abstract*: In this project a binary hypothesis test for the detection of uncovered background and moving pixels between image frames in a noisy image sequence has been formulated and evaluated. White Gaussian noise has been assumed. I have extended the binary hypothesis test to 3-ary hypothesis test, such that, the three segmentation regions are uncovered background, stationary and moving pixels. I have formulated the Bayes decision rule using a single intensity-difference measurement at each pixel. I have quantitatively evaluated the detection algorithm on an image sequence which I have generated.**

***Index Terms*- uncovered background, Bayes criterion, moving pixels, hypothesis testing**

I. INTRODUCTION

Literature in the area of image motion estimation contains a variety of techniques concerning this important issue. Image frames are generated by scanning a scene several times a second. Each frame consists of two parts: previous frame and moving part, which has changed from previous frame. Lot of research is going on in this particular area. Applications of this problem exist in the areas of data compression, filtering, video teleconferencing etc. [1].

The existence of newly introduces pixels, such as uncovered background pixels, in an image frame degrade the quality of the motion-compensated reconstructed image frames. Segmentation of an image frame in a sequence of images into regions of uncovered background, stationary and moving pixels play an important role in uncovered background prediction and motion compensation for image sequence coding. Many research papers suggest change detection method for the uncovered-background detection, but such methods are sensitive to noise.

In this paper, I present a method for segmenting an image into regions of pixels that correspond to those pixels which are displaced from the previous frame, i.e. moving pixels, those pixels which are the same as in the previous frame, i.e. stationary pixels, and those pixels which are newly introduced uncovered background pixels. I have used binary and ternary hypothesis testing using Bayes decision criterion.

II. BACKGROUND

In this project, I have used Bayes decision criterion for both binary and ternary hypothesis testing. For Bayes detection, it is assumed that the probabilities of hypothesis are known. A cost is assigned to each possible decision. Since consequences of one decision are different from the ones of another decision. Baye's criterion is based on determining the decision rule such that the average cost is minimum. The average cost is also known as the risk. Likelihood ratio is independent of probability of hypothesis as well as the assigned costs. In case the cost assignments can not be made, we can choose, $C_{10} – C_{00} = C_{01} – C_{11}$ [4].

III. MATHEMATICAL FORMULATION

A general mathematical model of the image motion can written as,

$$z_1(k) = s(k) + w_1(k),$$

where k denotes the spatial location (x, y), x and y integer, of the pixel in the image frame. Whereas $z_1(k)$ and $s(k)$ are the noisy and noise-free intensities, respectively, and $w_1(k)$ is zero mean white Gaussian noise. Now the current frame, or displaced frame intensity can be modeled as,

$$z_2(k) = \begin{cases} s(k-d(k)) + w_2(k), & k \notin \gamma_b \\ b(k) + w_2(k), & k \in \gamma_b \end{cases}$$

where d(k) is a non-uniform displacement vector, b(k) is the intensity of the scene background and $w_2(k)$ is zero mean white gaussian noise. And γ_b is the region of uncovered background. It is assumed that there are no illumination changes, no camera motion, and no changes in image acquisition parameters such as camera focus etc. [2]

Assuming that d(k) is small enough such that the first-order approximation of s(k-d(k)) is valid. In this event the above defined expression becomes,

$$z_2(k) \approx s(k) – g^T(k)\, d(k) + w_2(k),$$

where $g(k) = [g_1(k), g_2(k)]^T$ is the gradient vector of the intensity of the previous frame. Now defining,

$$\xi(k) = z_2(k) – z_1(k),$$

and

$$w(k) = w_2(k) – w_1(k)$$

Thus we obtain,

$$\xi(k) = \begin{cases} -g^T(k)\, d(k) + w_2(k), & k \notin \gamma_b \; :H1 \\ b(k) – s(k) + w_2(k), & k \in \gamma_b \; :H0 \end{cases}$$

Where ^{T} denotes matrix transpose. H_1 and H_0 are the binary hypothesis, where H_1 indicates a correspondence to a stationary (d(k)=0) or moving (d(k)≠0) pixel from the previous frame, and H_0 indicates the correspondence to uncovered background pixels.

Thus the likelihood ratio for binary hypothesis test can be formed as follows,

$$\Lambda[\xi(k)] = \frac{p(\xi(k)|H_1)}{p(\xi(k)|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \eta = \frac{P_0}{P_1}$$

Where $p(\xi(k)|H_1)$ and $p(\xi(k)|H_0)$ are the conditional probability density functions of ξ (k) given H_1 and H_0, respectively, and η is a threshold value [5]. Assuming b(k) and s(k) are statistically independent [2], we can write the likelihood ratio as,

$$\Lambda[\xi(k)] = \frac{\exp\left\{\frac{1}{2} m^T C\, m\right\} I(\xi(k))}{2\pi|K_d|^{1/2}\int \exp\left\{\frac{-2\xi(k)q(k)+q^2(k)}{2\sigma_w^2}\right\} p(q(k))\, dq}$$

Where,

$$q(k) = b(k) - s(k),$$

p(q(k) is the pdf of q(k), and σ_w^2, K_d, m, C, and $I(\xi(k))$ are as in [2].

Further I have extended the model for the current frame to a 3-ary hypothesis test to separate the non-background pixels into moving and stationary pixels. This can be written as follows,

$$z_2(k) = \begin{cases} s(k-d(k)) + w_2(k), & k \in \gamma_m \\ s(k) + w_2(k), & k \in \gamma_s \\ b(k) + w_2(k), & k \in \gamma_b \end{cases}$$

where γ_s, γ_m and γb are the regions of stationary, moving and uncovered background pixels, respectively. Similarly, defining $\xi(k)$ and w(k) as in the binary case, the three hypothesis becomes,

$$\xi(k) = \begin{cases} -g^T(k)d(k) + w_2(k), & k \in \gamma_m : H_0 \\ w(k), & k \in \gamma_s : H_1 \\ b(k) - s(k) + w_2(k), & k \in \gamma_b : H_2 \end{cases}$$

From these expressions, I have obtained following two likelihood ratios,

$$\Lambda_1[\xi(k)] = \frac{f[\xi(k)|H_1]}{f[\xi(k)|H_0]} = \frac{2\pi|K_d|^{1/2}}{\exp\left\{\frac{1}{2} m^T C\, m\right\} I(\xi(k))}$$

$$\Lambda_2[\xi(k)] = \frac{f[\xi(k)|H_1]}{f[\xi(k)|H_0]} = \frac{1}{\Lambda[\xi(k)]}$$

I have used the 3-ary hypothesis test as given in [2]. Note that $\Lambda_2[\xi(k)]$ is the inverse of the likelihood ratio developed for the binary hypothesis case as described earlier. Ternary hypothesis test using these expressions can be shown in (1) at the bottom the page.

I have used the intensity histogram of the moving object and the intensity histogram of the background to estimate p[q(k)]. I have performed the ternary hypothesis test using a single sample, the intensity-difference at the spatial location of the pixel I am classifying.

IV. Performance Evaluation

In order to evaluate the segmentation algorithm, several sets of experiments are performed. Several test image sequences consisting of several moving objects and different background scenes are generated. The images were used at several different signal-to-noise ratios using a single measurement for classifying each pixel [4]. The signal to noise ratio (SNR) can be defined as,

$$SNR = 10\log_{10}\left(\frac{\textit{average image variance}}{\textit{noise variance}}\right) dB$$

For the binary hypothesis test, I have generated receiver operating characteristic (ROC) curve at several SNR's. Whereas for the 3-ary hypothesis test, confusion matrix has been formed for the 20 dB SNR test case. The matrix shows the percentage of pixels of a given class that were detected in each of the three classes. The confusion matrix has been formed for several different cost matrices.

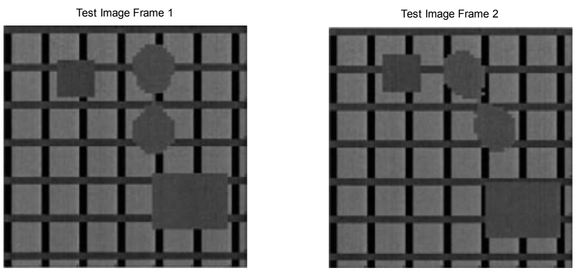
All of the above tests require knowledge of the pdf of the difference between background and object intensity, i.e. p[q(k)], in the region of uncovered background, which is

$$\begin{aligned} P_1(C_{01}-C_{11})\Lambda_1[\xi(k)] &\underset{H_0 \text{ or } H_2}{\overset{H_1 \text{ or } H_2}{\gtrless}} P_1(C_{01}-C_{11}) + P_2(C_{01}-C_{11})\Lambda_2[\xi(k)] \\ P_2(C_{02}-C_{22})\Lambda_2[\xi(k)] &\underset{H_0 \text{ or } H_2}{\overset{H_1 \text{ or } H_2}{\gtrless}} P_0(C_{20}-C_{00}) + P_1(C_{21}-C_{01})\Lambda_1[\xi(k)] \\ P_2(C_{12}-C_{22})\Lambda_2[\xi(k)] &\underset{H_0 \text{ or } H_2}{\overset{H_1 \text{ or } H_2}{\gtrless}} P_0(C_{20}-C_{10}) + P_1(C_{21}-C_{11})\Lambda_1[\xi(k)] \end{aligned} \quad (1)$$

determined by convolving the histogram of the background with the flipped histogram of the object and normalizing to make the result valid pdf.

V. Results & Discussion

I have used two image frames as shown in Fig. 1 to evaluate binary and ternary hypothesis test.



(a) (b)

Fig. 1 Generated test image sequence. (a) Noise-free previous frame. (b) Noise-free present frame.

Figure 2 shows the true segmentation of the image frame with white representing moving pixels, black representing stationary pixels, and gray representing uncovered background pixels.

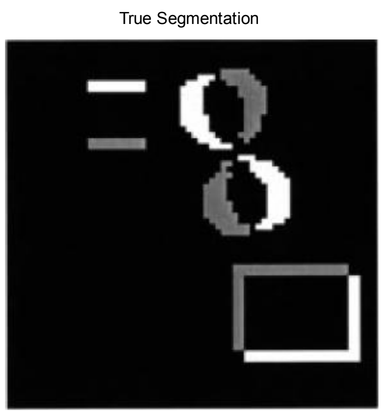
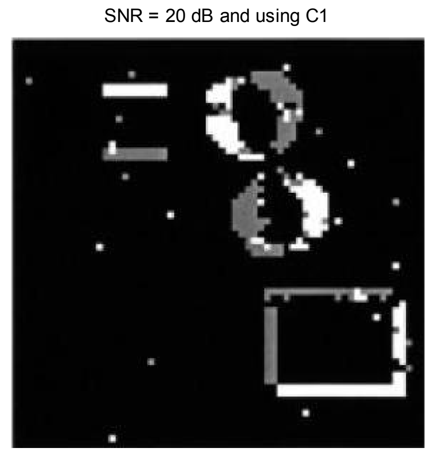


Fig. 2. True Segmentation. Black: Stationary pixels, White: moving pixels and Gray: uncovered background pixels.

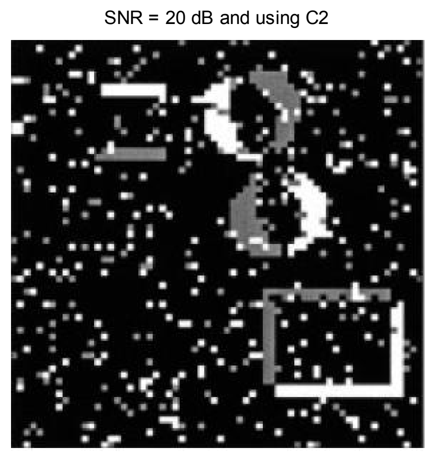
The cost values are application dependent and this it reflect the tolerance for misclassification in various regions. The result of the decision test depends on the accuracy of this *a priori* information. In figure 3, I have shown the segmentation resulting from hypothesis testing for image frames with an SNR of 20 dB and for two different cost matrices,

$$\mathbf{C_1} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \qquad \mathbf{C_2} = \begin{pmatrix} 0 & 1 & 1 \\ 9 & 0 & 9 \\ 1 & 1 & 0 \end{pmatrix}$$

and the probability matrix $P = [0.1 \quad 0.8 \quad 0.1]$. From the figure 3, we can see that if we make it more costly to misclassify a moving or uncovered background pixel as a stationary pixel, we increase the probability of detection of uncovered background and moving pixels, but we also misclassify many of the stationary pixels.



(a)



(b)

Fig. 3 Resulting segmentations using single measurement at SNR of 20 dB. (a) C_1 and P. (b) C_2 and P. Black: stationary pixels, White: moving pixels and Gray: uncovered background pixels.

Figure 4 shows the ROC curves for several SNR's and single measurement.

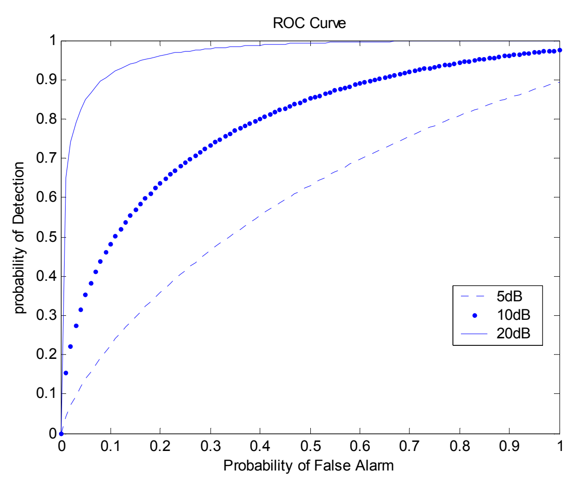


Fig. 4. ROC single measurement.

For ternary hypothesis test, the corresponding confusion matrices for the test case of 20 dB are shown in Tables I and II, respectively. From these tables, we see the effect of the cost matrix on the performance of the hypothesis test.

TABLE I CONFUSION MATRIX, IN PERCENTAGE, FOR TERNARY HYPOTHESIS TEST, SNR = 20 DB AND USING C_1

Detected Region	True Region		
	Moving	Stationary	Uncovered background
Moving	84	1	6
Stationary	8	98	13
Uncovered background	8	1	81

TABLE II CONFUSION MATRIX, IN PERCENTAGE, FOR TERNARY HYPOTHESIS TEST, SNR = 20 DB AND USING C_2

Detected Region	True Region		
	Moving	Stationary	Uncovered background
Moving	89	6	7
Stationary	3	89	4
Uncovered background	8	5	89

VI. SUMMARY AND CONCLUSIONS

In this paper, I have presented a binary and ternary hypothesis test, based on Bayes decision criteria, for the detection of moving, stationary and uncovered background pixels in image sequences. The segmentation results were presented only on the base of hypothesis test.

Future work:

This method can be further tested for real time teleconferencing image sequences. Active contour model algorithm (i.e. snake algorithm) can be used to segment the person contour in an image frame. Signal can be detected in presence of various types of noises such as colored noise.

VII. REFERENCE

[1] Zivkovic, Z.; Petkovic, M.; van Mierlo, R.; van Keulen, M.; van der Heijden, F.; Junker, W.; Rijnierse, E.; "Two video analysis applications using foreground/background segmentation"; Visual Information Engineering, 2003. VIE 2003. International Conference on , 2003 Pages:310 – 313

[2] "Non-uniform image motion estimation using Kalman filtering", N. M. Namazi, P. Penafiel, and C. M. Fan, IEEE Trans. Image Processing, Vol. 3, pp 191-212, 1989.

[3] "On regularization, formulation and initialization of the active contour models (snakes)", K. F. Lai and R. T. Chin, in Asian conf. Computer vision, Osaka, Japan, Nov. 1993, pp 542-545

[4] "Detection Theory – Application and Digital Signal Processing", R. Hippenstiel, CRC Press, 2002

[5] ECE 8433 – Class Notes.

Appendix

(Matlab code used in this project)

Program:

```
clc; clear all; close all;

% Loading the frames.
i1 = rgb2gray(imread('pic1','jpg'));
i2 = rgb2gray(imread('pic2','jpg'));

% Displaying the image sequence.
figure; imshow(i1);
title('Test Image Frame 1');
figure; imshow(i2);
title('Test Image Frame 2');

% Another sequence of frames for testing.
i11 = im2double(imread('1','jpg'));
i22 = im2double(imread('2','jpg'));

i1 = im2double(i1);
i2 = im2double(i2);

% Adding noise to the frames.
i1n = imnoise(i1,'gaussian',0,0.02);
i2n = imnoise(i2,'gaussian',0,0.02);
siz = size(i1);
w1 = randn(288, 289);
w2 = randn(288, 289);
w = w2 - w1;
i1nn = i1 + w1;
i2nn = i2 + w2;

% Displaying the given image sequences with noise added.
figure; imshow(i1n);
title('Noisy Image Frame 1');
figure; imshow(i2n);
title('Noisy Image Frame 2');

% Performing hypothesis testing.
z1 = i1nn;
dk = i2;
z2 = (i1-dk) + w2;
zheta = z2 - z1;

%under H1
[px, py] = gradient(i1);
temp = (px'*dk);
zhetaH1 = -temp(1:288,1:289) + w;
%under H0
zhetaH0 = w;

% Probability and cost functions.
P = [0.1 0.8 0.1];
C1 = [0 1 1; 1 0 1; 1 1 0];
C2 = [0 1 1; 9 0 9; 1 1 0];

var_zheta = (std2(zheta))^2
var_w = (std2(w))^2
var_d = (abs(var_zheta-var_w))/((norm(px))^2)
dxy = std2(i2);
kd = [var_d dxy; dxy var_d]
```

```
gamma = (px*px')/var_w;
m = (px/var_w)*zheta';

out = zeros(siz(1), siz(2));
for i=1:siz(1)
    for j=1:siz(2)
        if(i1(i,j)==i2(i,j))
            out(i,j)=1;
        end
    end
end
figure;
imshow(out)
figure; imshow(imread('result','jpg'))
title('True Segmentation');

siz1 = size(i11);
out = zeros(siz1(1), siz1(2));
for i=1:siz1(1)
    for j=1:siz1(2)
        if(i11(i,j)==i22(i,j))
            out(i,j)=1;
        end
    end
end

% Checking the inequalities condition.
lrt1 = (exp(0.5*m'*gamma*m)/(2*pi*sqrt(det(kd))));
LHS1 = P(2)*(C1(1,2)-C1(2,2))*lrt1;
RHS1 = P(1)*(C1(2,1)-C1(1,1)) + P(3)*(C1(2,3)-C1(1,3))*lrt1;
LHS2 = P(3)*(C1(1,3)-C1(3,3))*lrt1;
RHS2 = P(1)*(C1(3,1)-C1(1,1)) + P(2)*(C1(3,2)-C1(1,2))*lrt1;
LHS3 = P(3)*(C1(2,3)-C1(3,3))*lrt1;
RHS3 = P(1)*(C1(3,1)-C1(2,1)) + P(2)*(C1(3,2)-C1(2,2))*lrt1;

% Computing Signal to Noise Ratio in dB.
SNR = -10*log10((std2(i1))^2/var_w)
n1 = imread('n20dbc1','jpg');
figure; imshow(n1);
title('SNR = 20 dB and using C1');
n2 = imread('n20dbc2','jpg');
figure; imshow(n2);
title('SNR = 20 dB and using C2');

% ROC curve
snr=[.5 1 2];
sign = [':' '.-' '-'];
var = var_zheta;
figure;
for i=1:3
p_da = [];
for p_faa = 0:0.01:1
    x = 2*erfcinv(p_faa*sqrt(2));
    prob_da = 0.5*erfc(x/var-snr(i));
    p_da = [p_da prob_da];
end
p_faa = linspace(0,1,101);
plot(p_faa,p_da,sign(i))
hold on;
xlabel('Probability of False Alarm');
```

```
ylabel('probability of Detection');
title('ROC Curve');
legend('5dB','10dB','20dB',0)
end
hold off;

% Another image sequence.
figure; imshow(i11);
title('Image Frame 1');
figure; imshow(i22);
title('Image Frame 2');
figure; imshow(out);
title('Segmentation');
```