

PENDETEKSI OBJEK YANG DICURIGAI SEBAGAI BOM MENGUNAKAN SURVEILLANCE CAMERA

(Detecting Alleged Bomb Object Using Surveillance Camera)

Jony Kosasih*, Lukas**

**Fakultas Teknik Jurusan Teknik Elektro
Universitas Katolik Indonesia Atma Jaya – Jakarta
*jonyiverson@yahoo.co.id, **lukas@atmajaya.ac.id**

Abstrak

Tindakan terorisme semakin banyak terjadi. Bom meledak di tempat umum, seperti hotel, pusat perbelanjaan, dan bandara. Hal tersebut membuat masyarakat khawatir apabila berpergian ke tempat umum. Salah satu cara mengurangi tindakan terorisme adalah dengan memeriksa barang bawaan pengunjung saat ingin memasuki tempat umum. Namun, hal tersebut tidak cukup untuk menjaga keamanan tempat umum karena penjaga memiliki keterbatasan dalam pemeriksaan. Salah satu solusi terhadap hal ini adalah meletakkan kamera pengintai untuk mengamati daerah ramai pengunjung apabila terjadi hal yang mencurigakan. Untuk memaksimalkan fungsi kamera pengintai perlu dibuat sistem pendeteksi barang yang mencurigakan. Untuk mendapatkan latar belakang digunakan metode *Time Average Background Image* (TABI). Pengurangan piksel antara latar belakang dengan citra yang ditangkap oleh kamera menghasilkan objek yang terdeteksi. Objek tersebut akan dilacak dan ditandai dengan kotak warna hijau. Objek ditandai kotak warna kuning apabila dalam kondisi diam dan objek akan ditandai dengan kotak warna merah, ditampilkan pada jendela bahaya. Selanjutnya, sirine dibunyikan apabila objek tetap tidak bergerak. Berdasarkan pengujian, objek berhasil dideteksi dan ditandai oleh kotak dengan warna yang sesuai dengan penilaian.

Kata Kunci: pendeteksi barang yang mencurigakan, kamera pengintai, pengolahan citra

Abstract

Terrorisms more frequently occur today. Bombs exploded in public places such as hotels, shopping centers, and airports. Bomb threat concerns people traveling to public places. One way to reduce terrorism is by inspecting visitors' bags before entering public places. Considering the limitation of security guards do inspection, this method does not ensure security. One solution is to place surveillance cameras to observe crowded areas. To maximize the surveillance camera function, a detection system to scan suspicious objects was designed. Time Average Background Image (TABI) method was used to get the background. Reducing the pixel between the background image and the captured image resulted in detection of object. The object was tracked and marked a green box. The object was marked yellow box when stationary and red box to alert a danger. The siren was activated when the object remains motionless. The experiment showed successful detection of objects.

Keywords: suspicious items detection, surveillance camera, image processing

Tanggal Terima Naskah : 8 Maret 2013
Tanggal Persetujuan Naskah : 18 Maret 2013

1. PENDAHULUAN

1.1 Latar Belakang

Tindakan terorisme semakin meresahkan masyarakat. Banyak terjadi bom meledak di tempat umum, seperti hotel, pusat perbelanjaan, dan bandara. Banyaknya ancaman bom semakin membuat orang kuatir untuk bepergian ke tempat umum karena merasa tidak aman. Untuk menjaga keamanan sebagian besar tempat umum menggunakan penjaga di pintu masuk untuk memeriksa apakah terdapat barang yang dicurigai sebagai bom yang dibawa oleh pengunjung. Memeriksa pengunjung tidak cukup untuk menjaga keamanan tempat umum dikarenakan penjaga juga memiliki keterbatasan dalam memeriksa.

Pada beberapa negara maju digunakan kamera untuk merekam aktivitas di suatu daerah. Kamera dapat mendeteksi apabila terjadi hal yang mencurigakan. Dengan kemajuan teknologi, kamera yang berfungsi sebagai pengambil gambar dapat juga digunakan sebagai pengawas. Dengan mengambil hasil rekaman dan diolah oleh komputer, hasil rekaman tersebut dapat memberi peringatan apabila terdapat barang yang mencurigakan.

Untuk meminimalkan terjadinya kejahatan yang tidak diinginkan, dibutuhkan sistem keamanan dengan menggunakan kamera pengintai yang dapat mendeteksi objek yang dicurigai sebagai bom dan menunjukkan letak objek yang mencurigakan sehingga pengguna dengan cepat dapat memeriksa barang tersebut.

1.2 Tujuan

Penelitian ini bertujuan untuk merancang dan membuat program untuk kamera keamanan yang berfungsi sebagai pendeteksi barang yang dicurigai sebagai bom.

1.3 Pembatasan Masalah

Pembatasan masalah pada penelitian ini adalah:

- *Webcam* tidak bergerak hanya mengamati suatu daerah tertentu.
- Terdapat pencahayaan pada daerah yang diamati.

2. KONSEP PENGOLAHAN CITRA

2.1 Definisi

Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Proses pengolahan citra dalam bentuk digital secara umum mempertimbangkan masalah peningkatan mutu citra atau perbaikan citra. Jika mendapatkan suatu gambar yang terlalu gelap, dengan pengolahan citra gambar tersebut dapat diproses agar terlihat lebih jelas [1].

Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh *bit* tertentu [2]. Gambar 1 menunjukkan diagram blok pengolahan citra.



Gambar 1. Diagram blok pengolahan citra

2.2 Citra Digital

Sebuah citra harus diubah menjadi bentuk digital agar dapat diolah. Untuk membuat citra digital, diperlukan perubahan data kontinu menjadi data digital. Hal ini meliputi dua proses, yaitu *sampling* dan kuantisasi. Proses perubahan citra ke bentuk digital dapat menggunakan berbagai perangkat, misalnya *webcam*, *scanner*, *handycam*, kamera digital, dan perangkat keras lainnya.

Sebuah citra digital dapat didefinisikan sebagai fungsi dua dimensi $f(x,y)$, dengan x dan y adalah koordinat spasial dan amplitudo dari f pada masing-masing koordinat (x,y) yang disebut intensitas atau *level* keabuan pada koordinat citra tersebut [3]. Ketika x,y dan amplitudo dari f semuanya terbatas maka citra tersebut sebagai citra digital.

Citra dapat direpresentasikan dalam bentuk matriks. Misalkan citra berukuran $M \times N$ dimana M merupakan baris dan N merupakan kolom, maka representasi citra dapat ditunjukkan pada persamaan (1).

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1, N-1) \end{bmatrix} \dots\dots\dots(1)$$

Setiap elemen dalam matriks disebut elemen citra atau piksel. Dalam hal ini, $f(x,y)$ menunjukkan intensitas citra sedangkan x dan y menunjukkan posisi piksel dalam citra atau disebut koordinat spasial citra.

2.3 Citra RGB

Citra RGB disebut juga citra *true color*. Citra RGB merupakan citra digital yang mengandung matriks data berukuran $M \times N \times 3$ yang merepresentasikan warna merah, hijau, dan biru untuk setiap pikselnya. Setiap warna dasar diberi rentang nilai. Rentang nilai paling kecil 0 dan paling besar 255 untuk citra RGB 24 bit (*true color*). Setiap piksel warna merah, hijau, dan biru masing-masing memiliki 8 bit data sehingga total sebuah piksel citra RGB adalah 24 bit dan total warna yang dapat diperoleh lebih dari 16 juta warna. Warna dari tiap piksel ditentukan oleh kombinasi dari intensitas merah, hijau, dan biru.

2.4 Citra Keabuan

Citra keabuan (*grayscale image*) adalah suatu citra dengan nilai dari setiap piksel merupakan derajat intensitas. Citra yang ditampilkan dengan keabuan terdiri dari warna hitam pada bagian yang intensitas terlemah, warna putih pada intensitas terkuat, dan warna abu-abu pada intensitas antara hitam sampai dengan putih.

Citra keabuan dapat dihasilkan dengan mengkonversi citra RGB. Konversi dapat dilakukan dengan mengambil nilai rata-rata dari nilai *red* (R), *green* (G), dan *blue* (B). Konversi citra RGB menjadi citra keabuan dapat dituliskan dengan persamaan (2).

$$I(i,j) = \frac{R(i,j) + G(i,j) + B(i,j)}{3} \dots\dots\dots(2)$$

Keterangan:

- $I(i,j)$ = intensitas pada piksel dengan posisi i,j
- $R(i,j)$ = intensitas pada piksel merah dengan posisi i,j
- $G(i,j)$ = intensitas pada piksel hijau dengan posisi i,j
- $B(i,j)$ = intensitas pada piksel biru dengan posisi i,j

Gambar 2 menunjukkan hasil konversi citra RGB menjadi citra keabuan.



Gambar 2. Konversi citra RGB menjadi citra keabuan

2.5 Citra Biner

Citra biner adalah citra keabuan yang hanya memiliki dua derajat keabuan, yaitu putih dan hitam. Citra biner telah melalui proses pemisahan piksel-piksel berdasarkan derajat keabuan yang dimiliki. Pembentukan citra biner memerlukan nilai batas keabuan yang akan digunakan sebagai nilai patokan. Piksel dengan derajat keabuan lebih besar dari nilai batas akan diberi nilai 1 dan sebaliknya piksel dengan derajat keabuan lebih kecil dari nilai batas akan diberi nilai 0 [2]. Gambar 3 menunjukkan konversi citra keabuan menjadi citra biner.



Gambar 3. Konversi citra keabuan menjadi citra biner

2.6 Rekonstruksi Latar Belakang

Latar belakang adalah benda-benda yang berada di belakang objek inti dari suatu citra. Latar belakang merupakan citra yang paling sering terlihat dari serangkaian gambar bergerak. Latar belakang memiliki intensitas yang stabil pada waktu yang lama. Latar belakang pada serangkaian gambar bergerak dapat diperoleh dengan cara menangkap citra yang pertama kali atau dengan cara mengambil nilai rata-rata dari serangkaian gambar. Nilai rata-rata tersebut dapat mendekati gambar latar belakang yang diinginkan. Metode ini sudah umum dipakai dengan menggunakan istilah *time average background image* (TABI) [3].

TABI merupakan metode yang digunakan untuk menghasilkan gambar latar belakang dengan cara mendapatkan nilai rata-rata nilai piksel gambar dari serangkaian *frame*. Untuk menghitung TABI dapat digunakan persamaan (3).

$$\bar{F}(x,y) = \frac{\sum_{i=1}^n f_i(x,y)}{n} \dots\dots\dots (3)$$

Keterangan:

$\bar{F}(x,y)$ = rata-rata piksel pada posisi x,y

$f_i(x,y)$ = intensitas piksel pada *frame* i pada posisi x,y

n = jumlah *frame*

i = urutan *frame*

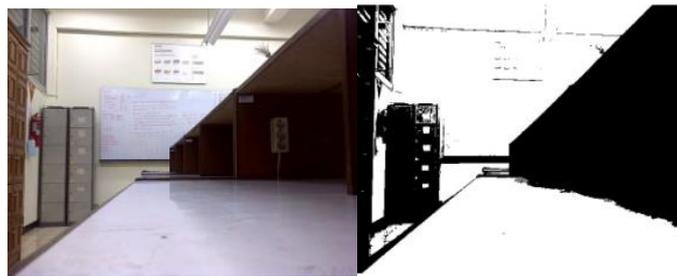
2.7 Deteksi Objek Bergerak

Objek bergerak adalah suatu objek warna, posisi, ukuran, atau tempat dalam *frame* citra yang berubah berdasarkan waktu. Dalam suatu rangkaian citra akan mendapatkan objek yang bergerak apabila terdapat perubahan nilai setiap piksel pada koordinat yang sama antara *frame* sekarang dengan *frame* sebelumnya [4].

Untuk mendapatkan perubahan nilai setiap piksel dapat menggunakan metode pengurangan latar belakang. Metode ini bertujuan untuk memisahkan objek dengan latar belakang sehingga gerakan objek terdeteksi. Metode pengurangan latar belakang dapat dihitung dengan persamaan (4).

$$\text{Pengurangan} = | \text{latar belakang} - \text{latar depan} | \dots\dots\dots (4)$$

Latar belakang adalah citra referensi yang bisa diperoleh melalui proses TABI sedangkan latar depan adalah citra pembanding yang diperoleh dari setiap *frame* yang ingin diproses. Sebelum melakukan metode pengurangan latar belakang, latar belakang dan latar depan terlebih dahulu harus diubah menjadi citra biner. Setelah itu, proses pengurangan latar belakang dapat dilakukan untuk mendapatkan citra yang baru. Apabila pada citra baru didapatkan warna hitam (bernilai 0) maka nilai dan posisi piksel antara citra pembanding dengan citra referensi sama sedangkan pada citra baru didapatkan warna putih (bernilai 1) maka ada perubahan nilai dan posisi piksel antara citra pembanding dengan citra referensi atau dapat dikatakan terdeteksi pergerakan. Untuk lebih jelasnya dapat melihat Gambar 4, Gambar 5, dan Gambar 6.



Gambar 4. Latar belakang dan latar belakang dalam bentuk biner



Gambar 5. Latar depan dan latar depan dalam bentuk biner



Gambar 6. Hasil subtraksi latar belakang

2.8 Bounding Box

Bounding box adalah proses yang dilakukan setelah segmentasi, yaitu dengan menggunakan pengurangan latar belakang [5]. Gambar divisualisasikan dengan *bounding box* melalui dimensi objek yang dihasilkan dari gambar sebelumnya. Untuk gambar tertentu, *scanning* dilakukan dengan nilai intensitas gambar lebih dari batas (tergantung pada nilai yang diberikan, untuk menetapkan keakuratan yang maksimum). Dalam hal ini, fitur diekstraksi oleh warna yang menggambarkan nilai intensitas warna. Nilai-nilai intensitas dari atas, bawah, kiri, dan kanan menggunakan dimensi ini. Nilai kotak pembatas persegi panjang dihubungkan dalam batas-batas nilai yang dihasilkan [3]. Gambar 7 menunjukkan contoh dari *bounding box*.



Gambar 7. Contoh *bounding box*

3. PERANCANGAN SISTEM

3.1 Uraian Proses

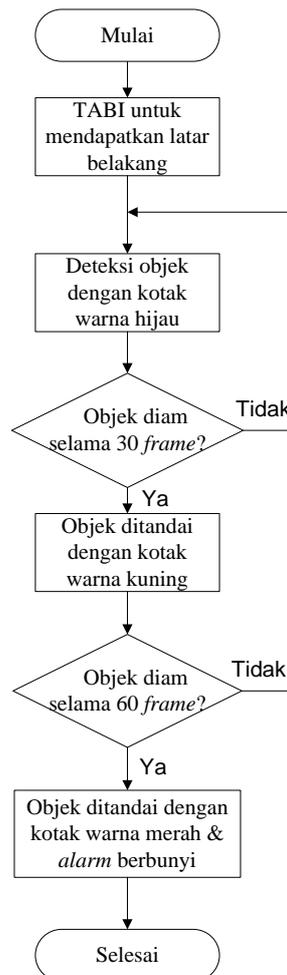
Sistem yang dibuat memiliki tiga tingkatan tanda objek yang dicurigai sebagai bom. Tingkatan pertama adalah amati. Objek pada tingkat pertama ditandai dengan kotak warna hijau. Setiap objek yang terdeteksi akan ditandai dengan kotak warna hijau. Tingkatan kedua adalah waspada. Pada tingkatan kedua objek yang terdeteksi didapati diam selama beberapa detik maka objek akan ditandai dengan kotak warna kuning. Objek yang ditandai dengan kotak warna kuning patut diwaspadai oleh pengamat. Tingkatan ketiga adalah bahaya. Pada tingkat ketiga objek didapati terus-menerus diam melebihi waktu yang ditentukan akan ditandai dengan kotak warna merah disertai bunyi *alarm*. Jika objek sudah ditandai dengan kotak warna merah disertai bunyi *alarm* diharapkan pengamat menindaklanjuti dan memeriksa objek tersebut [6].

Pertama, aktifkan sistem dengan menekan tombol *start*. *Web camera* akan merekam secara *real time*. Hasil rekaman akan diproses di komputer dalam bentuk *frame*. Untuk mendapatkan hasil yang akurat dibutuhkan model latar belakang secara terus-menerus. Latar belakang didapatkan dengan menggunakan metode TABI. Untuk jumlah *frame* yang akan digunakan dalam metode TABI bisa diatur oleh pengguna. Pada sistem digunakan 100 *frame* untuk metode TABI [7]. Setelah mendapat latar belakang, *web camera* akan melanjutkan merekam untuk mendapatkan latar depan. Setelah mendapat citra RGB yang merupakan latar depan selanjutnya akan dilakukan proses *thresholding* dan proses pengurangan latar belakang. Pengurangan latar belakang merupakan proses pengurangan nilai piksel antara latar belakang dengan latar depan untuk mendapatkan objek yang diam. Selanjutnya dilakukan proses morfologi citra terhadap citra yang dihasilkan dari pengurangan latar belakang. Proses morfologi citra meliputi erosi, *filling*, *thicken*, dan dilasi. Setelah proses morfologi selesai masuk ke proses selanjutnya, yaitu *labelling* yang berfungsi memberikan label pada objek yang terhubung dalam citra. Pada

objek yang sudah diberi label akan dilakukan *scanning* untuk menentukan titik tengah dan titik *bounding box* pada objek. Selanjutnya, titik tengah objek yang diperoleh digunakan untuk proses pelacakan dan titik *bounding box* digunakan untuk menampilkan *kotak* berwarna hijau.

Letak objek hasil pelacakan pada *frame* saat ini akan dibandingkan dengan letak objek hasil pelacakan pada *frame* sebelumnya. Apabila terdapat kesamaan letak maka parameter penentu objek ditandai dengan kotak warna kuning akan bertambah 1. Nilai parameter tersebut akan terus bertambah apabila terdapat kesamaan letak terus-menerus sehingga nilai melebihi batas yang ditentukan (sebagai contoh pada sistem ditentukan 30 *frame*), maka objek tersebut akan ditandai dengan berubahnya warna kotak dari hijau menjadi kuning. Warna kuning menandakan objek tersebut patut diwaspadai oleh pengamat. Jika pada saat merekam objek tersebut berpindah letaknya maka parameter penentu objek ditandai kotak warna kuning akan kembali bernilai 1 dan ditandai dengan kotak warna hijau.

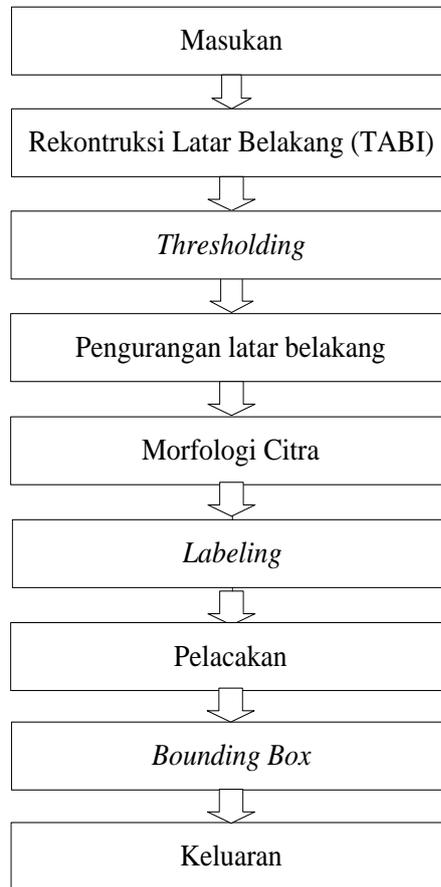
Objek yang ditandai dengan kotak warna kuning yang berarti waspada dapat menjadi bahaya apabila letak objek hasil pelacakan pada *frame* saat ini yang dibandingkan dengan letak objek hasil pelacakan pada *frame* sebelumnya didapati nilai parameter melebihi nilai yang ditentukan (sebagai contoh pada sistem ditentukan 60 *frame*) maka objek tersebut akan ditandai dengan kotak berwarna merah disertai bunyi *alarm* tanda benda tersebut patut dicurigai. Gambar 8 menampilkan diagram alir sistem yang dirancang.



Gambar 8. Diagram alir sistem

3.2 Diagram Blok Sistem

Sistem yang dibuat akan dibagi menjadi beberapa bagian, yaitu masukan, rekonstruksi latar belakang (TABI), *thresholding*, pengurangan latar belakang, morfologi citra, *labelling*, pelacakan, *bounding box*, dan keluaran. Gambar 9 menunjukkan diagram blok sistem secara berurutan.



Gambar 9. Diagram blok sistem

3.3 Masukan

Masukan yang digunakan pada sistem adalah rekaman objek secara *real time*. *Framerate* dan jumlah *frame* digunakan untuk menentukan berapa lama objek diam sedangkan resolusi gambar video digunakan untuk melakukan keseluruhan proses sistem. Apabila terjadi perbedaan resolusi gambar video yang akan diproses maka sistem tidak akan berjalan dengan baik. Resolusi gambar video pada sistem ini adalah 320 x 240 piksel. Pada sistem diatur setiap *trigger* menghasilkan 1 *frame*, jumlah *trigger* tidak terbatas, dan untuk melakukan *trigger* ditentukan secara manual. Berikut ini adalah kode pada MATLAB untuk masukan.

```
vid = videoinput('winvideo', 1, 'RGB24_320x240');  
set(vid, 'FramesPerTrigger', 1);  
set(vid, 'TriggerRepeat', Inf);  
triggerconfig(vid, 'Manual');
```

3.4 Rekontruksi Latar Belakang

Rekontruksi latar belakang diperoleh dengan menggunakan metode TABI. Langkah-langkah untuk mendapatkan latar belakang menggunakan metode TABI dengan cara menjumlahkan setiap piksel dari serangkaian *frame*. Setelah itu, jumlah piksel yang didapat dibagi oleh banyaknya *frame* sehingga diperoleh rata-rata nilai dari piksel. Metode TABI banyak menghasilkan derau yang ditandai dengan kemunculan kelompok piksel kecil dengan ukuran kecil.

Penerapan TABI untuk video benda yang dicurigai sebagai bom yang memiliki 100 *frame* diperoleh latar belakang yang sudah cukup baik. Untuk objek dideteksi sebagai latar belakang asumsi kemunculan objek di atas 50% dari jumlah *frame*. Berikut ini adalah kode pada MATLAB untuk rekonstruksi latar belakang dengan metode TABI.

```
Imzero=double(zeros(240,320,3));
for i = 1:frmback
    trigger(vid);
    Im{i} = double(getdata(vid,1));
    Imzero = Im{i}+Imzero;
end
Imback = Imzero/frmback;
```

3.5 Thresholding

Thresholding dilakukan untuk mendapatkan citra biner. Citra biner akan digunakan pada proses pengurangan latar belakang. Keadaan lingkungan tempat objek memiliki intensitas cahaya yang berbeda. Misalnya pada ruangan terbuka di siang hari intensitas cahaya akan berbeda pada ruang tertutup dengan pencahayaan yang dihasilkan oleh lampu. Untuk mendapatkan nilai *thresholding* yang sesuai dengan keadaan lingkungan objek yang dideteksi terlebih dahulu diambil sepertiga dari nilai rata-rata piksel dari latar belakang. Berikut ini adalah kode pada MATLAB untuk melakukan *thresholding*.

```
thresh = sum(sum(Imback));
RED = thresh(:, :, 1) / (76800*3);
GREEN = thresh(:, :, 2) / (76800*3);
BLUE = thresh(:, :, 3) / (76800*3);
```

3.6 Pengurangan Latar Belakang

Citra yang didapat dari proses perekam secara *real time* menghasilkan citra RGB. Untuk melakukan proses pengurangan latar belakang citra RGB harus diubah menjadi citra biner yang dilakukan melalui proses *thresholding* sebelumnya. Pengurangan latar belakang digunakan untuk mendapatkan objek yang diam dengan melakukan proses pengurangan nilai piksel antara latar belakang dengan latar depan. Berikut ini adalah kode pada MATLAB untuk pengurangan latar belakang

```
fore = (abs(Imfore(:, :, 1, i) - Imback(:, :, 1)) > RED) ...
    | (abs(Imfore(:, :, 2, i) - Imback(:, :, 2)) > GREEN) ...
    | (abs(Imfore(:, :, 3, i) - Imback(:, :, 3)) > BLUE);
```

Citra yang didapat dari pengurangan latar belakang akan menghasilkan piksel-piksel kecil yang akan mengurangi ketelitian objek yang terdeteksi. Untuk memperbaiki citra tersebut dibutuhkan proses morfologi citra.

3.7 Morfologi Citra

Hasil dari pengurangan latar belakang masih memiliki *noise* berupa piksel-piksel kecil yang dapat mengurai ketelitian objek yang diam. Untuk menghilangkan *noise* maka dilakukan proses morfologi citra pada gambar yang dihasilkan dari proses subtraksi latar belakang. Terdapat empat proses yang akan dilakukan dalam morfologi citra, yaitu erosi, *filling*, *thickening*, dan dilasi. Proses ini dilakukan secara berurutan.

3.7.1 Erosi

Proses erosi berfungsi mengecilkan atau menipiskan objek pada citra biner. Dalam sistem pendeteksi objek yang dicurigai sebagai bom, erosi digunakan untuk menghapus *noise* berupa piksel kecil yang dapat mengurangi ketelitian dalam mendeteksi objek. Dalam MATLAB untuk menjalankan proses erosi digunakan sintaks *bwmorph*. Pada sintaks *bwmorph* terdapat banyak pilihan operasi, seperti *thicken* dan dilasi yang digunakan dalam sistem. Untuk melakukan proses erosi digunakan operasi dalam *bwmorph*, yaitu *erode*. Kode pada MATLAB untuk melakukan erosi adalah:

```
Erode = bwmorph(fore, 'erode', 2);
```

3.7.2 Filling

Proses erosi yang berguna untuk menghapus *noise* yang berupa piksel kecil menyebabkan terjadi pelebaran lubang pada objek yang terdeteksi. Untuk menutup lubang yang disebabkan oleh proses erosi supaya objek terdeteksi dengan baik digunakan proses *filling*. Proses *filling* digunakan untuk mengisi lubang-lubang pada citra biner. Lubang yang bisa diisi adalah satu set piksel citra yang tidak dapat dicapai dengan mengisi citra dari tepi gambar. Pada MATLAB digunakan sintaks *imfill*. Untuk melakukan *filling* digunakan kode pada MATLAB sebagai berikut.

```
fill = imfill(erode, 'holes');
```

3.7.3 Thickening

Untuk *Thickening* digunakan sintaks *bwmorph* dengan pilihan operasinya adalah *thicken*. Kode pada MATLAB untuk *thickening* sebagai berikut.

```
Thick=bwmorph(fill, 'thicken', 1);
```

Proses *thickening* digunakan untuk mempersiapkan citra biner untuk proses dilasi. Proses *thickening* berfungsi menebalkan setiap area tanpa menghubungkan daerah yang sebelumnya tidak terhubung.

3.7.4 Dilasi

Proses dilasi berfungsi untuk melakukan penumbuhan pada citra biner. Proses dilasi dilakukan untuk mengembalikan citra biner seperti awal sebelum dilakukan erosi untuk menghilangkan *noise* berupa piksel kecil. Pada MATLAB untuk melakukan proses *thickening* digunakan sintaks *bwmorph* dengan pilihan operasinya adalah *thicken*. Berikut ini adalah kode pada MATLAB untuk melakukan dilasi.

```
dilate=bwmorph(thic, 'dilate', 2);
```

3.8 Labelling

Setelah proses morfologi citra dilakukan selanjutnya adalah proses *labelling*. Proses *labelling* berfungsi memberikan label pada objek yang terdeteksi. Pada proses *labelling* di MATLAB menggunakan sintaks *bwlabel*.

```
labeled = bwlabel(dilate,8);
```

3.9 Pelacakan

Pada sistem deteksi objek yang dicurigai sebagai bom untuk melakukan pelacakan objek digunakan metode Kalman *Filter*. Setiap objek yang dilacak dinyatakan berdasarkan titik tengah (*centroid*) dari setiap objek yang terdeteksi. Metode Kalman *Filter* menggunakan informasi dari objek terdeteksi pada suatu *frame* dan status objek tersebut di *frame* sebelumnya untuk membuat suatu perkiraan dari status baru objek tersebut. Sebelum proses pelacakan dilakukan terlebih dahulu harus ditetapkan *filter*-nya. Kode pada MATLAB untuk *filter* Kalman seperti berikut.

```
R=[ [0.2845,0.0045] ', [0.0045,0.0455] '];
H=[ [1,0] ', [0,1] ', [0,0] ', [0,0] '];
Q=0.01*eye(4);
P = 100*eye(4);
dt=1;
A=[ [1,0,0,0] ', [0,1,0,0] ', [dt,0,1,0] ', [0,dt,0,1] '];
g = 6;
Bu = [0,0,0,g] ';
```

Setelah ditetapkan tapis (*filter*) yang akan digunakan selanjutnya dilakukan proses pelacakan. Untuk melacak setiap objek yang terdeteksi digunakan titik tengah setiap objek. Berikut ini adalah kode pada MATLAB untuk melacak objek menggunakan titik tengah.

```
for a = 1:N
xp{: ,a}=A*x{: ,a}(i,:) ' + Bu
end
PP = A*P*A' + Q;
K = PP*H'*inv(H*PP*H'+R);
P = (eye(4)-K*H)*PP;
for a = 1:N
x{: ,a}(i,:) = (xp{: ,a} + K*([cc{: ,a},cr{: ,a}] ' -
H*xp{: ,a})) ' ;
end
```

Hasil pelacakan nantinya akan digunakan untuk mendeteksi apakah terdapat objek yang diam. Cara mendeteksi objek yang diam adalah dengan membandingkan nilai hasil pelacakan *frame* saat ini dengan nilai hasil pelacakan pada *frame* sebelumnya. Jika terdapat kesamaan nilai pelacakan dengan nilai toleransi 60 maka parameter objek yang terdeteksi sebagai bom akan bertambah satu. Parameter digunakan untuk mengetahui tingkatan objek yang terdeteksi. Parameter akan terus bertambah 1 apabila objek yang terdeteksi menghasilkan nilai perbandingan yang terus-menerus sama. Jika nilai parameter mencapai lebih dari 30 maka warna kotak pada objek yang terdeteksi akan menjadi kuning yang berarti objek tersebut harus diwaspadai. Bila nilai parameter mencapai lebih dari 60 maka warna kotak pada objek yang terdeteksi akan berubah

menjadi warna merah, ditampilkan secara khusus pada layar monitor, dan sirine dibunyikan. Berikut ini adalah kode pada MATLAB untuk menandai objek dengan kotak warna kuning apabila diam sebanyak 30 *frame* dan kotak warna merah apabila diam sebanyak 60 *frame*.

```
if abs(x{:,a}(i,1:4)-x{:,a}(i-1,1:4)) < 60
    detect{:,a} = detect{:,a} +1
else
    detect{:,a} = 1
end

if detect{:,a} > yellow

rectangle('Position',bounding{a},'EdgeColor','yellow','LineW
idth',2)
end

if detect{:,a} > red

rectangle('Position',bounding{a},'EdgeColor','red','LineWidt
h',2)
    detect{:,a} = 1;
    pause(0.1)
    crop{a} = imcrop(Im{i}, bounding{a});
    imshow(uint8(crop{a}));

system('"C:\Program Files\Winamp\winamp.exe" /PLAY
"E:\Music\siren.mp3" &')
```

3.10 *Bounding Box*

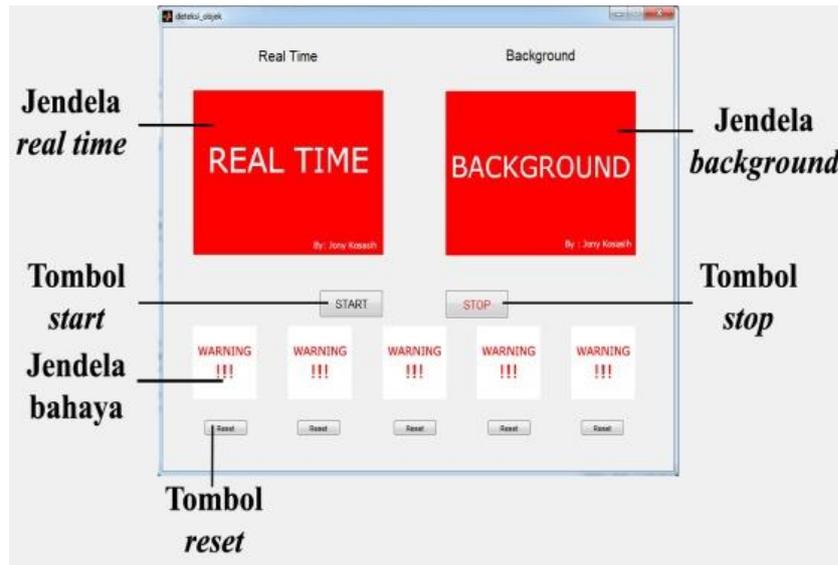
Setelah objek yang diam sudah terdeteksi, pemakai harus mengetahui dimana letak objek tersebut. Proses *bounding box* dilakukan supaya pemakai sistem dapat mengetahui dimana letak objek yang dicurigai sebagai bom. Terdapat tiga warna *bounding box* sesuai dengan tingkatan objek yang dicurigai sebagai bom. Warna pertama adalah hijau apabila objek dalam tingkat amati, warna kedua adalah kuning apabila objek dalam tingkat waspada, dan warna ketiga adalah merah apabila objek dalam tingkat bahaya. Pada MATLAB untuk menampilkan *bounding box* digunakan sintaks *rectangle*.

4. PENGUJIAN SISTEM

4.1 Pengujian Pada Ruang Tertutup

Tujuan dari pengujian dari ruang tertutup adalah untuk mengetahui apakah sistem dapat mendeteksi objek dengan baik pada keadaan ruangan yang memiliki pencahayaan merata dan tetap ke seluruh daerah pengujian. Pengujian dilakukan di Unika Atma Jaya, Jalan Jend. Sudirman 51, Jakarta. Sebagai contoh ruang tertutup yang ada di Unika Atma Jaya digunakan lantai bawah PKPM Gedung K2 sebagai tempat pengujian. Gambar 12 merupakan denah lokasi pengujian pada ruang tertutup.

Pada awal program dijalankan akan muncul jendela utama sistem yang dapat dilihat pada Gambar 10.



Gambar 10. Jendela utama sistem

Untuk memulai menjalankan program klik tombol *start* pada jendela utama. Setelah program dijalankan, kamera akan mulai merekam dan hasil rekaman akan ditampilkan pada jendela *real time*. Program akan mengambil sebanyak 100 *frame* untuk dijadikan latar belakang dengan metode TABI dan latar belakang hasil dari metode TABI akan ditampilkan pada jendela *background*. Proses pengambilan latar belakang dengan TABI berjalan dengan baik. Hal tersebut dapat dilihat pada *frame* ke-50 sampai *frame* ke-75 terdeteksi objek berupa orang sedang menaiki tangga tetapi hasil latar belakang dengan proses TABI objek tersebut tidak dijadikan bagian dari latar belakang. Gambar 11 merupakan tampilan beberapa *frame* untuk mendapatkan latar belakang, dan tampilan setelah didapati latar belakang dapat dilihat pada Gambar 12.



a) *Frame* ke-1



b) *Frame* ke-50

Orang sedang menaiki tangga



c) *Frame* ke-75

Orang sedang menaiki tangga

Gambar 11. Tampilan beberapa *frame* untuk mendapatkan latar belakang



Gambar 12. Tampilan setelah didapati latar belakang

Setelah didapatkan latar belakang program mulai mendeteksi objek yang dicurigai sebagai bom. Hasil deteksi akan ditampilkan pada jendela *real time*. Semua objek yang terdeteksi akan ditandai dengan kotak warna hijau. Objek ditandai dengan kotak warna kuning apabila objek terdeteksi diam selama 30 *frame*. Objek ditandai dengan kotak warna merah, ditampilkan pada jendela bahaya dan sirine dibunyikan apabila objek terdeteksi diam selama 60 *frame*. Gambar 13 merupakan tampilan objek yang terdeteksi diam sebanyak 30 *frame* dan objek yang terdeteksi diam sebanyak 60 *frame* dapat dilihat pada Gambar 14.



Gambar 13. Tampilan objek yang terdeteksi diam sebanyak 30 *frame*



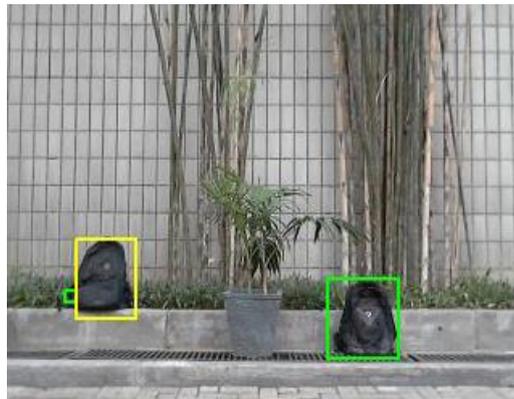
Gambar 14. Tampilan objek yang terdeteksi diam sebanyak 60 *frame*

Pada pengujian di ruang tertutup objek yang diberi pencahayaan yang merata dan tetap oleh lampu dapat terdeteksi dengan baik. Hanya saja terdapat bayangan dari objek yang terdeteksi oleh sistem sehingga kotaknya melebihi objek yang terdeteksi. Untuk sistem penilaian objek yang berbahaya berjalan dengan baik. Objek yang diam selama 30 *frame* ditandai dengan kotak warna kuning, objek yang diam selama 60 *frame* ditandai dengan kotak warna merah dan objek ditampilkan pada jendela bahaya. Secara garis besar sistem bekerja dengan baik pada ruang tertutup.

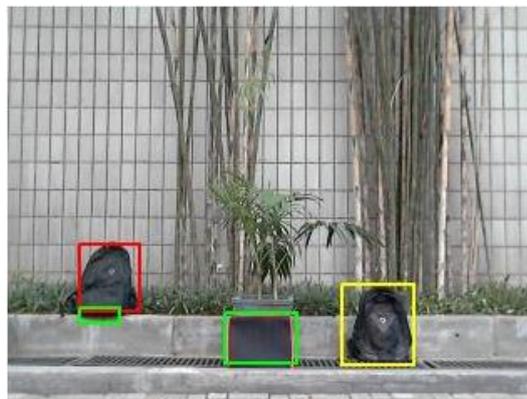
4.2 Pengujian Ruang Terbuka

Tujuan dari pengujian dari ruang terbuka adalah untuk mengetahui apakah sistem dapat mendeteksi objek dengan baik pada keadaan ruangan yang memiliki pencahayaan oleh matahari yang berubah dan tidak merata ke seluruh daerah pengujian. Sebagai contoh tempat pengujian ruang terbuka digunakan halaman terbuka yang berada di dekat Hall B Unika Atma Jaya. Percobaan dilakukan pada pukul 16.00.

Pengujian sama seperti pada ruang tertutup. Seratus *frame* pertama digunakan untuk mendapatkan latar belakang. Selanjutnya, objek yang diam selama 30 *frame* akan ditandai dengan kotak warna kuning. Objek yang diam selama 60 *frame* akan ditandai dengan kotak warna merah, ditampilkan pada jendela bahaya, dan sirine dibunyikan. Gambar 15 merupakan tampilan objek yang terdeteksi diam sebanyak 30 *frame* dan Gambar 16 menunjukkan tampilan objek yang terdeteksi diam sebanyak 60 *frame*.



Gambar 15. Tampilan objek yang terdeteksi diam sebanyak 30 *frame*



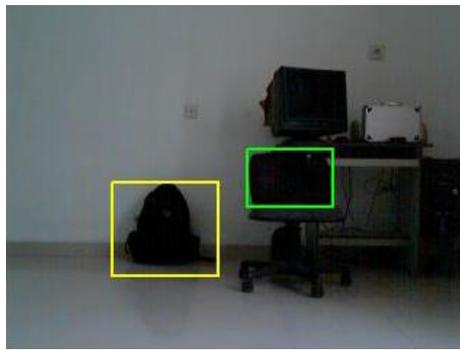
Gambar 16. Tampilan objek yang terdeteksi diam sebanyak 60 *frame*

Pada pengujian di ruang terbuka objek yang diberi pencahayaan oleh matahari yang berubah dan tidak merata dapat terdeteksi dengan baik. Objek dapat ditandai kotak dengan baik sesuai dengan ukuran objek, hanya saja terdapat piksel-piksel kecil yang terdeteksi sebagai objek oleh sistem. Hal ini karena pencahayaan di ruang terbuka yang tidak merata sehingga program menjadikan cahaya tersebut sebagai sebuah objek. Untuk sistem penilaian objek yang berbahaya juga bekerja dengan baik. Objek yang diam selama 30 *frame* ditandai dengan kotak warna kuning dan objek yang diam selama 60 *frame* ditandai dengan kotak warna merah dimana objek ditampilkan pada jendela bahaya. Dengan demikian dapat ditunjukkan sistem bekerja dengan baik pada ruang terbuka.

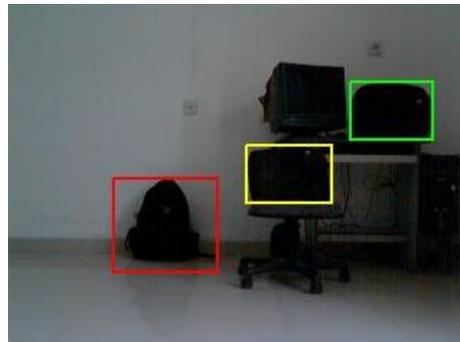
4.3 Pengujian pada Ruang Tertutup dengan Intensitas Cahaya yang Kurang/Redup

Tujuan dari pengujian dari ruang tertutup dengan intensitas cahaya yang kurang/redup adalah untuk mengetahui apakah objek dapat terdeteksi dengan baik pada ruang tertutup yang diberi pencahayaan yang berasal dari matahari melalui jendela. Ruangan tertutup dengan intensitas cahaya yang kurang/redup menggunakan sebuah kamar tidur. Pengujian dilakukan pada pukul 16.00.

Pengujian sama seperti pada ruang tertutup dan ruang terbuka. Objek yang diam selama 30 *frame* akan ditandai dengan kotak warna kuning. Objek yang diam selama 60 *frame* akan ditandai dengan kotak warna merah, ditampilkan pada jendela bahaya, dan sirine dibunyikan. Gambar 17 merupakan tampilan objek yang terdeteksi diam sebanyak 30 *frame* dan Gambar 18 merupakan tampilan objek yang terdeteksi diam sebanyak 60 *frame*.



Gambar 17. Tampilan objek yang terdeteksi diam sebanyak 30 *frame*



Gambar 18. Tampilan objek yang terdeteksi diam sebanyak 60 *frame*

Pada pengujian dari ruang tertutup dengan intensitas cahaya yang kurang/redup objek dapat terdeteksi dengan baik. Objek ditandai dengan kotak sesuai dengan ukuran objek. Untuk sistem penilaian objek yang berbahaya berjalan dengan baik juga. Objek yang diam selama 30 *frame* ditandai dengan kotak warna kuning dan objek yang diam selama 60 *frame* ditandai dengan kotak warna merah disertai objek ditampilkan pada jendela bahaya. Dengan demikian dapat ditunjukkan sistem bekerja dengan baik pada ruang tertutup dengan intensitas cahaya yang kurang/redup.

4.4 Pengujian Pada Keadaan Ramai

Tujuan dari pengujian pada keadaan ramai adalah untuk mengetahui apakah objek dapat terdeteksi dengan baik pada keadaan ramai. Keadaan ramai adalah keadaan yang terdapat banyak orang lalu-lalang pada daerah tangkap kamera. Sebagai contoh

tempat dengan keadaan ramai yang digunakan pada pengujian adalah lantai 3 di *Margo City*, Depok.

Pada pengujian keadaan ramai akan diperlihatkan beberapa hasil *print screen* pada saat pengujian beserta analisis pada keadaan tersebut.



Gambar 19. Hasil *print screen* ke-1

Pada Gambar 19 terdapat dua objek yang terdeteksi berupa orang yang sedang berdiri. Objek yang terdeteksi tidak terlalu baik karena hanya separuh bagian dari objek ditandai dengan kotak. Hal tersebut dikarenakan baju yang dipakai oleh objek yang terdeteksi berwarna putih sehingga pada saat proses *thresholding* objek menjadi sama dengan latar belakang mendapat nilai 1 (berwarna putih) pada citra biner. Selain itu, terdapat objek yang sedang berjalan tidak terdeteksi oleh sistem. Hal tersebut dikarenakan objek menggunakan baju berwarna putih.

Pada hasil *print screen* ke-1 dapat dianalisis bahwa objek yang berwarna cerah seperti putih tidak dapat dideteksi oleh sistem dengan baik. Hal tersebut dikarenakan pada saat proses *thresholding* pengubahan citra RGB menjadi citra biner objek yang berwarna cerah dikonversi menjadi nilai 1 (putih) sehingga saat dikurangi dengan latar belakang yang juga bernilai 1, objek menjadi bernilai 0 (hitam) yang sama seperti latar belakang.



Gambar 20. Hasil *print screen* ke-2

Pada Gambar 20 terdeteksi dua objek yang merupakan orang sedang berdiri. Objek terdeteksi dengan baik, terdapat objek yang ditandai dengan kotak warna kuning dan objek yang ditandai dengan kotak warna hijau. Objek ditandai dengan kotak warna kuning karena objek diam selama 30 *frame*. Objek ditandai dengan kotak warna hijau karena objek tersebut baru terdeteksi diam tidak lebih dari 30 *frame*.



Gambar 21. Hasil *print screen* ke-3

Pada Gambar 21 terdapat objek yang terdeteksi berupa orang yang sedang diam. Objek tersebut ditandai dengan kotak warna merah dan ditampilkan pada jendela bahaya karena sudah diam selama 60 *frame*. Selain itu, pada jendela *real time* terdapat objek yang sedang berdiri yang merupakan penjaga toko akan tetapi tidak terdeteksi oleh sistem karena terlalu jauh letaknya sehingga sistem tidak dapat mendeteksi objek tersebut.

5. KESIMPULAN

Berdasarkan pada hasil pengujian yang telah dilakukan, maka didapatkan kesimpulan sebagai berikut:

- 1) Sistem penilaian objek yang berbahaya berjalan dengan baik. Objek yang diam selama 30 *frame* akan ditandai dengan kotak warna kuning dan untuk 60 *frame*, objek akan ditandai dengan kotak warna merah, ditampilkan di jendela bahaya dan sirine dibunyikan.
- 2) Pencahayaan mempengaruhi hasil deteksi objek, apabila pencahayaan tetap dan merata ke seluruh daerah pengamatan akan menghasilkan deteksi objek yang baik.
- 3) Objek yang berwarna cerah seperti putih tidak dapat dideteksi oleh sistem dengan baik dikarenakan pada saat proses *thresholding* pengubahan citra RGB menjadi citra biner objek yang berwarna cerah dikonversi menjadi nilai 1 (putih) sehingga saat dikurangi dengan latar belakang yang juga bernilai 1, objek bernilai 0 (hitam) menjadi sama seperti latar belakang.
- 4) Untuk mendapatkan hasil deteksi yang baik dibutuhkan pencahayaan yang tetap dan merata.

REFERENSI

- [1]. Shinta, E. P., "Aplikasi Webcam untuk Mendeteksi Gerakan Suatu Objek," Makalah Tugas Akhir, Fakultas Teknik Elektro Universitas Diponegoro, Semarang, 2010.
- [2]. Prasetyo, E., "Pengolahan Citra Digital dan Aplikasinya Menggunakan MATLAB," Penerbit ANDI, Yogyakarta, 2011.
- [3]. Hartoto, P., "Sistem Deteksi Kecepatan Kendaraan Bermotor pada Real Time Traffic Information System," Tugas Akhir, Fakultas Teknik Elektro Institut Teknologi Surabaya, Surabaya, 2011.
- [4]. Firdausy, K., Daryono, dan Yudhana, A., "Webcam untuk Sistem Pemantauan Menggunakan Metode Deteksi Gerakan," Skripsi, Fakultas Teknik Elektro Universitas Ahmad Dahlan, Yogyakarta, 2009.

- [5]. Acharya. T dan Ray. A. K., "*Image Processing: Principle and Applications*," John Wiley & Sons, Inc, New Jersey, 2005.
- [6]. Amrullah, R. Y., "*Pengenalan Benda di Jalan Raya dengan Metode Kalman Filter*." Fakultas Teknik Elektro Institut Teknologi Surabaya, Surabaya, 2010.
- [7]. Sigit, R., "*Step by Step Pengolahan Citra Digital*," ANDI, Yogyakarta, 2005.