



# **A Study on Image Change Detection Methods for Multiple Images of the Same Scene Acquired by a Mobile Camera**

**Guntur Tanjung**

School of Mechanical Engineering  
The University of Adelaide  
South Australia 5005  
Australia

*A thesis submitted in fulfilment of the requirements  
for the degree of Doctor of Philosophy in  
Mechanical Engineering  
on the 12<sup>th</sup> October 2009*

## CHAPTER 7

---

### **DETECTING OF OBJECTS ABSENCE AND PRESENCE BEHIND FENCE WIRES**

The complete outdoor change detection method is built from two distinctive algorithms. The first algorithm is used only to detect objects disappearing and appearing behind fence wires in multiple outdoor images of the same scene containing fence wires captured by a mobile camera from slightly different positions, angles and at different times. The second algorithm is used only to detect breaches in the integrity of and attached objects in front of fence wires in these kinds of multiple outdoor images and it will be presented in Chapter 9.

Chapter 7 describes only the first algorithm. All information in the previous chapters (i.e., Chapters 4 – 6) will be integrated together in developing the first algorithm. Section 7.1 begins with background of problems that arise from these kinds of multiple outdoor image of the same scene containing fence wires. Solutions of how to overcome problems mentioned in Section 7.1 are presented in Section 7.2. Section 7.3 presents changed masks produced by the first algorithm. Concluding remarks are presented in Section 7.4.

#### **7.1 Background**

When multiple images of the same scene are captured by a mobile camera from somewhat different positions, angles and at different times, the displacement of same objects observably appears in the multiple images, referred to as parallax. If the multiple images are flat plane images (i.e., images produced by Landsat multispectral scanner (MSS), Landsat thematic mapper (TM), magnetic resonance (MR) and positron emission tomography (PET)), the parallax can be fixed correctly by registering the multiple images. However, when multiple images of the same scene are depth plane images (i.e., images captured by a camera in such

a way that the camera position is perpendicular to the scene), registering the multiple images does not solve the parallax entirely. Pixels that are closer to control points, which are used to register the multiple images, have smaller disparities and pixels that are far from control points have larger disparities. Moreover, objects presence and absence in the multiple images are occluded regions in terms of stereo correspondence research. These disparities and occluded regions lead to an investigation of stereo correspondence algorithms that explicitly detect these occluded regions. Extracting these occluded regions can be used as potential significant changes in terms of image change detection. These occluded regions could still contain significant and unimportant changes. To determine which occluded regions belong to significant or unimportant changes, a first hybrid decision-making system is developed.

The complexity of change detection further increases when detecting regions of change is performed to multiple outdoor images of the same scene taken by a mobile camera since background clutter of the outdoor scene and illumination variation during the day have also to be included in consideration. Therefore, the first algorithm must be able to register these kinds of multiple outdoor images in reducing parallax, overcome background clutter of the outdoor scene, extract occluded regions in the reference and registered input images and decide which occluded regions belong to significant or unimportant changes. Moreover, it must be invariant towards illumination variation. The first algorithm is presented in the next section.

### **7.2 Algorithm Overview**

As known from the first conclusion in the initial study (see Section 3.5), parallax in multiple images of the same scene taken by a mobile camera can be minimized by registering both reference and current input images. Prior to register both reference and input images, distinctive correspondence points firstly have to extract automatically from both reference and input images. Extracting control points in these kinds of multiple outdoor images is a challenging task. There are no natural objects that can be used as distinctive objects in utilizing them as sources of control points since the outdoor scene contains a chain-link mesh

fence. Natural objects in this kind of outdoor scene are often less of features. Moreover, in these kinds of multiple outdoor images, a big natural object will be divided into several separated small objects by the chain-link mesh fence.

To overcome this difficulty, two same artificial templates (i.e., a building picture) were used in this research. Both templates were attached on left and right posts of the wire fence. The SIFT operator was chosen in this research as a local features extractor since local features provided by the SIFT operator (i.e., keypoints) are invariant to image scale and rotation, and are tolerant of small changes in illumination. The SIFT operator is suitable in this research since objects in these kinds of multiple outdoor images change in sizes and pixel values as a result of camera motion and illumination variation. A template-based matching approach was developed in the research in extracting control points from both reference and input images. The template-based matching approach and experimental results of applying it to multiple outdoor images of the same scene are described in Chapter 4. Output of the template-based matching approach is a couple of correspondence points extracted from both reference and input images.

Based on information of control points extracted in the previous process, the current input image was secondly registered to a same coordinate system with the reference image. The linear conformal transformation was chosen in this research in transforming the input image since there is no projection that occurs in the input image. As a result, there are two areas observably in the registered input image: a gray area with information on it and a black area with no information on it. Since the black area does not contain any information, it has to be cropped from the registered input image. As a consequence of black area cropping, size of the registered input image will significantly decrease. Since size of the registered input image must be same with size of the reference image in the next process, the reference image must be cropped as well by using information used to crop the registered input image. To extract automatically information used to crop the registered input and reference images, a simple region of interest extraction algorithm was developed in the research. A detail explanation of how to register both reference and input images and how to crop regions of interest from the reference and registered input images is presented in Chapter 5. Output of the

image registration was two regions of interest of both reference and registered input images (i.e., ROI2\_RI and ROI2\_II).

If both ROI2\_RI and ROI2\_II are subtracted, a few things can observe in the subtracted image. The things are that (1) left and right posts of the wire fence overlap, (2) fence wires have smaller gaps and (3) same objects behind fence wires have larger disparities. Left and right posts of the wire fence overlie because locations of correspondence points used to register both reference and input images are at the left and the right posts of the wire fence (i.e., zero disparities). Fence wires have smaller gaps because locations of control points are near left and right posts of the wire fence and same objects behind fence wires have larger disparities because locations of control points are far away from left and right posts of the wire fence. Moreover, appearing and disappearing of objects behind fence wires in the subtracted image can become occluded regions. In terms of stereo correspondence algorithms, occluded regions are regions that appear and/or disappear in the left image and disappear and/or appear in the right image. Zero, smaller and larger disparities that appear in the subtracted image lead on an investigation of stereo correspondence algorithms in detecting occluded regions in both ROI2\_RI and ROI2\_II. As a result of the investigation, the Zitnick and Kanade algorithm was chosen in the research since this algorithm can detect explicitly occluded regions in two stereo images.

The next process is to use both ROI2\_RI and ROI2\_II as input to the Zitnick and Kanade algorithm in generating a confidence map image. In the confidence map image, same objects (i.e., objects that exist in both left and right images and same objects have disparities) will be presented in lighter regions and occluded objects (i.e., objects that exist in the left image, but, they vanish in the right image, vice versa) will be presented in darker regions. In generating a confidence map image, the Zitnick and Kanade algorithm needs eight input parameters as mentioned in Section 6.2. Input parameters used by the Zitnick and Kanade algorithm during the experiment are summarized in Table 6.1. The set of input parameters is (0, 8, 1, 2, 1, 5, 0.96, SAD).

Values of 3<sup>rd</sup>, 4<sup>th</sup>, 5<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup> input parameters are chosen in this research since the values are recommended by Zitnick and Kanade (2000). For the 6<sup>th</sup> input

parameter, a number 5 is experimentally preferred since there are no significant changes any more on the confidence map image after five iterations.

Values of 0 and 8 are experimentally selected for the 1<sup>st</sup> and 2<sup>nd</sup> input parameters. Multiple outdoor images of the same scene containing fence wires often suffer from background clutter of the outdoor scene. Since the research focuses only on detecting presence and absence of objects behind fence wires (i.e., occluded regions and fence wires that have smaller disparities), not regions of change on the background (i.e., same objects that have larger disparities), the Zitnick and Kanade algorithm can be utilized only to search regions in certain range of disparities by limiting range of disparities (i.e., minimum and maximum disparity values in pixels). By choosing values of 0 and 8, the Zitnick and Kanade algorithm was utilized to search regions which have disparity values from 0 to 8 while disregarding regions which have disparity values greater than 8 pixels. By using these input parameters, the Zitnick and Kanade algorithm generates a confidence map image in which occluded regions are represented in darker regions and same objects that have smaller gaps are depicted in lighter regions in the confidence map image. The next challenge is to extract occluded regions from the confidence map image.

After investigation of pixel values of occluded regions in the confidence map image, pixels values of occluded regions tend to have intensity values in between 39 and 40. An intensity histogram of occluded regions was created and found that values of 39 and 40 appear as the first and second peak values in the histogram. The brightness thresholding was next applied to the confidence map image in which values of 39 and 40 were used as global threshold values. As a result, an occlusion map image was produced and introduced in this research. The occlusion map image only contains occluded regions extracted from the confidence map image. The complete explanation and results of confidence and occlusion map images making are described in Chapter 6.

To obtain a changed mask, there are two more steps performed to the confidence map image. The further steps are to decide which occluded regions in the confidence map image belong to significant or unimportant changes using a hybrid decision-making system and to reduce false positives in the image

generated by the hybrid decision-making system using a template-subtraction approach. Both the decision-making system and template subtraction approach will be described separately in next two sections.

### 7.3 First Hybrid Decision-Making System

As depicted in each occlusion map image (OMI<sub>II</sub>), see Figs. 6.8 – 6.12 in Section 6.4, sizes, shapes and positions of occluded regions that belong to significant changes are different with sizes, shapes and positions of occluded regions that belong to unimportant changes. To determine which occluded regions that belong to significant or unimportant changes in each OMI, a hybrid decision-making system was developed in this research. The term of “hybrid” is used since there are two different inference approaches combined together in the building the hybrid decision-making system. The approaches are a crisp inference system and a fuzzy inference system.

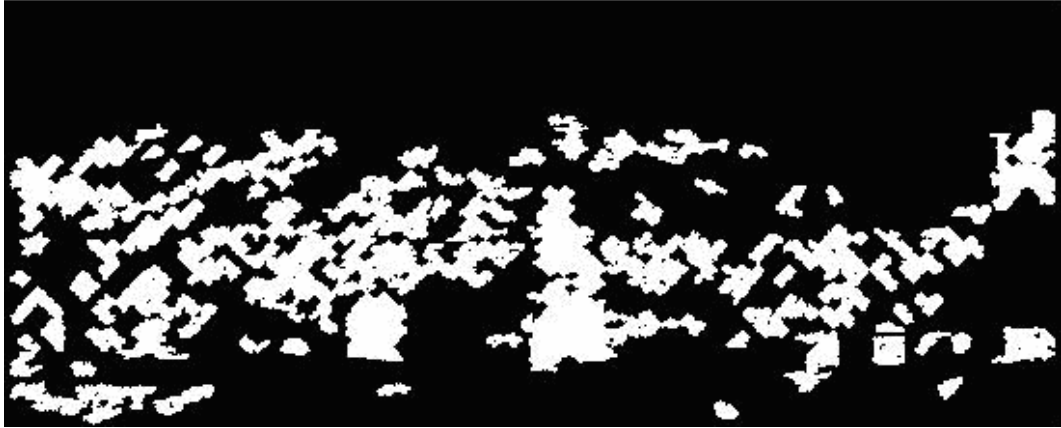
#### 7.3.1 First Crisp Inference System

The crisp inference system is utilized to remove occluded regions in each OMI based on their positions in Y axis. Since searching of significant changes is only limited from ground to 1.5 meters height, occluded regions positioned higher than 1.5 meters will be removed from each OMI by the crisp inference system. Occluded regions in 1.5 meters height represent the sky, tops of left and right posts of the wire fence and tops of trees and buildings. Equation (7.1), below, is used to remove occluded regions in each OMI based on information of their positions in Y axis.

$$B_5(x, y) = \begin{cases} B_4(x, y) & \text{if } Y_p > \eta \\ 0 & \text{otherwise} \end{cases} \quad (7.1)$$

where  $B_4(x, y)$  is the OMI,  $Y_p$  is the y axis position of each occluded region in a spatial space and  $\eta$  is a value set constant for any OMI at 370 pixels. The value of  $\eta$  is experimentally determined.

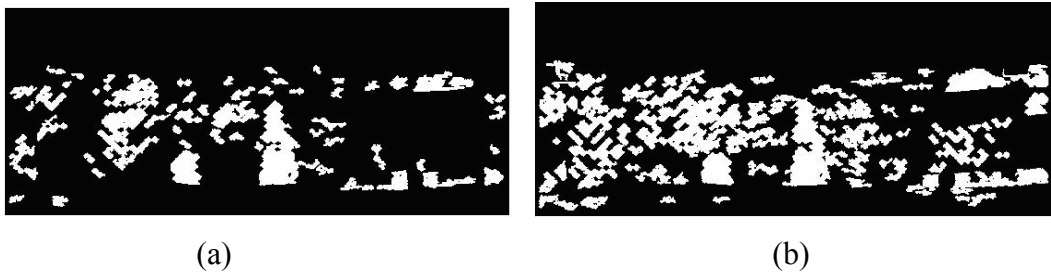
Next, the crisp inference system is applied to all OMI\_IIs. Fig. 7.1 depicts the result of applying the crisp inference system to the OMI\_II-1 (see Fig. 6.8 in Section 6.4).



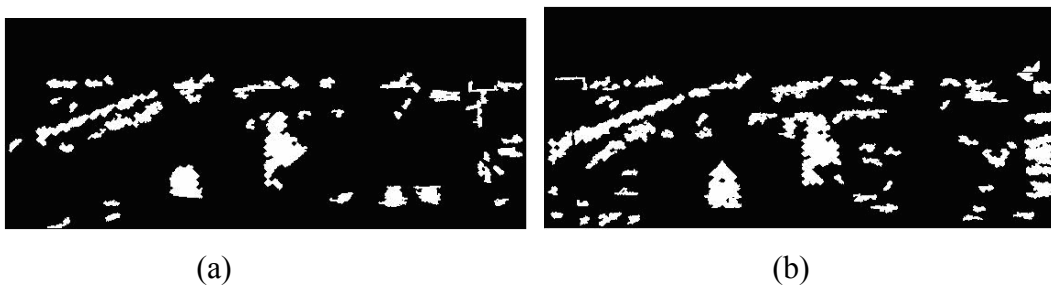
**Fig. 7.1** Removing occluded regions based on their positions in Y axis

As seen in Fig. 7.1, occluded regions, whose positions in Y axis are lower than 370 pixels, have been removed from the OMI\_II-1.

Figs. 7.2 (a) – (b), 7.3 (a) – (c), 7.4 (a) – (c) and 7.5 (a) – (c) depict results of applying the crisp inference system to other OMI\_IIs.



**Fig. 7.2** Results of applying the crisp inference system to the OMI\_II-5 (a) and OMI\_II-9 (b)

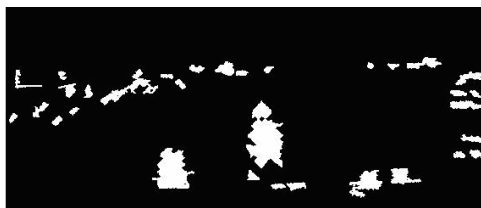




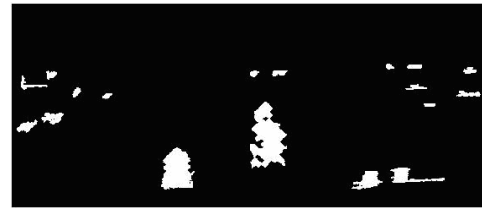


(c)

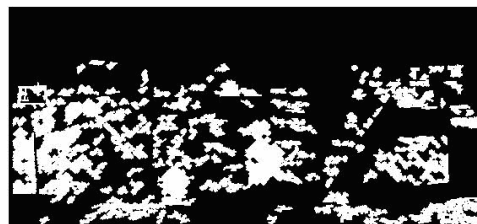
**Fig. 7.3** Results of applying the crisp inference system to the OMI\_II-2 (a), OMI\_II-6 (c) and OMI\_II-10 (c)



(a)



(b)



(c)

**Fig. 7.4** Results of applying the crisp inference system to the OMI\_II-3 (a), OMI\_II-7 (c) and OMI\_II-11 (c)



(a)



(b)



(c)

**Fig. 7.5** Results of applying the crisp inference system to the OMI\_II-4 (a), OMI\_II-8 (c) and OMI\_II-12 (c)

As seen in Figs. 7.2 (a) – (b), 7.3 (a) – (c), 7.4 (a) – (c) and 7.5 (a) – (c), above, any occluded regions, whose positions in Y axis are lower than 370 pixels have been eliminated. Moreover, occluded regions in the images above must be classified into two separated groups: significant and unimportant changes. To classify occluded regions, a first fuzzy inference system is developed and will be described in the next sub-section.

### 7.3.2 First Fuzzy Inference System (FIS1)

There are a few reasons of developing the first fuzzy inference system (FIS1) in this research. The first reason is that areas and shapes of occluded regions which belong to significant changes are quite different from areas and shapes of occluded regions which belong to unimportant changes. Areas and shapes of occluded regions that belong to significant changes tend to be bigger and to have specific contours such as rectangle, oval, square, and trapezoid. Although areas of occluded regions which belong to unimportant changes are occasionally bigger than areas of occluded regions which belong to significant changes, their shapes have random contours. The second reason is that areas and shapes of occluded regions which belong to significant changes always change because of camera motion. Changes in areas and shapes of occluded regions can be overcome by fuzzy rules of the fuzzy inference system since values of variables utilized in the fuzzy rules are represented in the closed interval of two specific numbers.

A Mamdani's fuzzy inference system (Asai and Aschmann, 1995; Sato *et al.*, 2000) was built in this study. An overview of the FIS1 is depicted in Fig. 7.6, below.  $A_r$ ,  $Y_w$  and  $P_b$  stand for area, y-width and probability. As seen in Fig. 7.6, below, the FIS1 consists of 2 inputs (area and y-width), 1 output (probability) and 5 rules. Area is interpreted as {small, average, large}, y-width is decoded as {small, large} and probability is translated as {low, high}. In addition, the y-width fuzzy input represents the height of an object in pixels.

The FIS1 was composed from five main steps:

1. **Fuzzify inputs.** The first step is to take the inputs (i.e., numerical values limited to the universe of discourse of input variables and outputs are fuzzy

degrees of membership in the qualifying linguistic set  $[0, 1]$ ) and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. In this study, the values between 0 to 200,000 and 0 to 1,264 are experimentally determined for input variables. These maximum values represent two and half times of an average area in pixels of adult sizes in the real life.

2. **Apply fuzzy operation (OR = max and AND = min).** Once the inputs have been fuzzified, the degree to which each part of the antecedent has been satisfied for each rule is known. If the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. This number is then applied to the output function. The input to the fuzzy operator is two or more membership values from fuzzified input variables. The output is a single truth value.
3. **Apply implication method (min).** The input for the implication process is a single number given by the precursor, and the output is a fuzzy set. Implication is implemented for each rule.
4. **Aggregate all outputs (max).** Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set. The input of the aggregation process is the list of truncated output functions returned by the implication process for each rule. The output of the aggregation process is one fuzzy set for each output variable.
5. **Defuzzify.** The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single number. The centroid defuzzification method was used in this study. It returns the center of area under the curve as the aggregate output fuzzy set. A single number of the defuzzification output represents a possibility percentage of an object ( $P$ ). The value of possibility percentage is in the range of 0 to 100 ( $[0, 100]$ ).

Next, the FIS1 is applied to the images in Figs. 7.1, 7.2 (a) – (b), 7.3 (a) – (c), 7.4 (a) – (c) and 7.5 (a) – (c). As a result of applying the FIS1 to every occluded region in the images, the FIS1 generates a possibility percentage value ( $P$ ). Equation (7.2), below, is utilized to determine whether or not an occluded regions

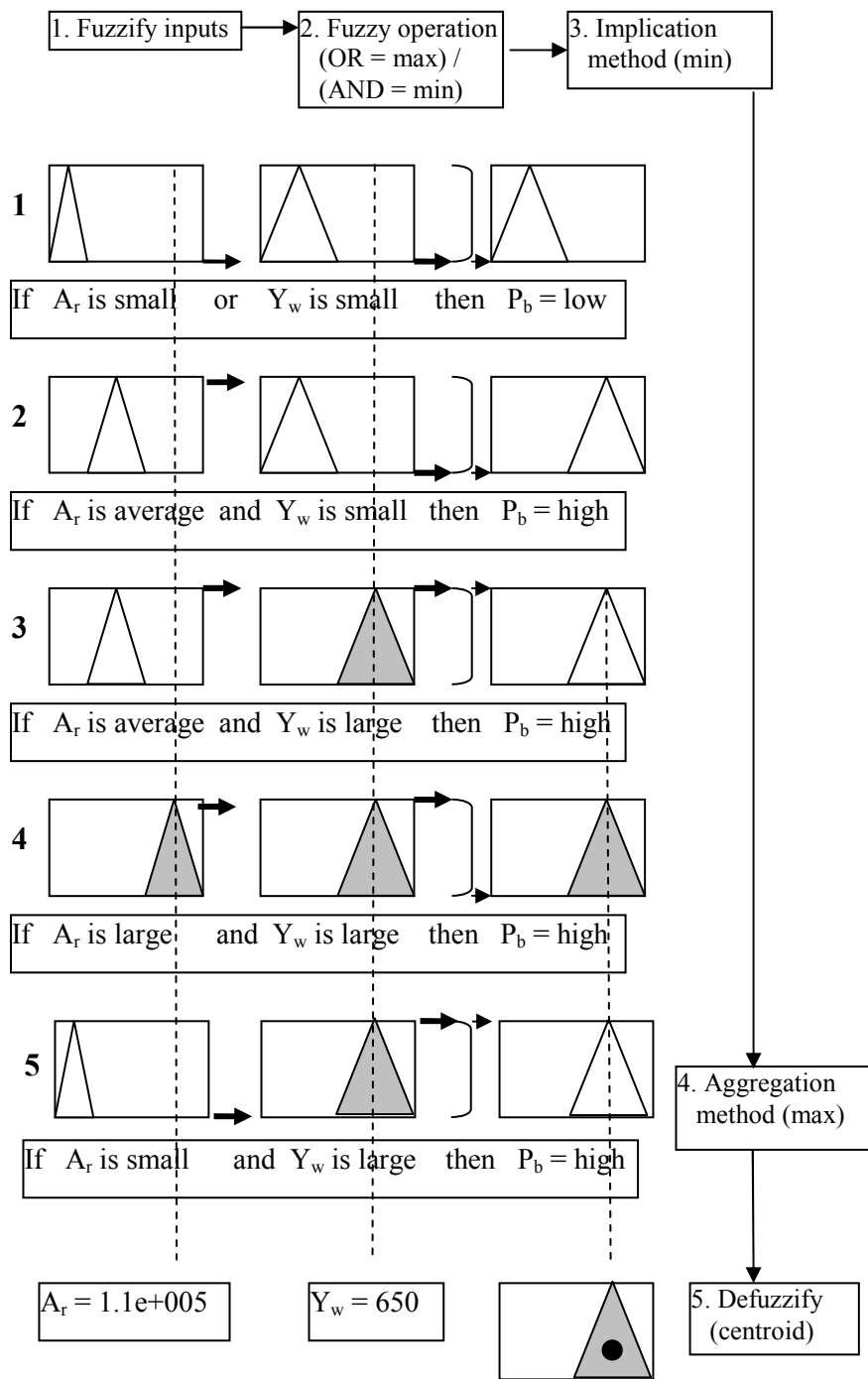


Fig. 7.6 The first fuzzy inference system (FIS1)

belongs to a significant change or an unimportant change based on information of its  $P$  value.

$$B_6(x, y) = \begin{cases} B_5(x, y) & \text{if } P \geq \rho \\ 0 & \text{otherwise} \end{cases} \quad (7.2)$$

where  $B_6(x, y)$  is a new binary image referred to as first hybrid decision-making system input image (HDS1\_II),  $P$  is the possibility percentage value of an occluded region and  $\rho$  is a value set constant for any OMI\_II at 49. In any OMI\_II, occluded regions consist of potential significant changes (i.e., a school bag and the intruder) and unimportant changes (i.e., fence wires, wall of the fence, objects in front of fence wires and objects are far away on the background). Occluded regions that belong to unimportant changes must be suppressed while occluded regions that belong to significant changes must be detected. Since areas and heights of occluded regions that belong to significant changes are quite different compared with occluded regions that belong to unimportant changes, areas and heights of occluded regions are then used as inputs to the FIS1. Thus, the value of 49 is experimentally determined and it represents an average value of areas and heights in Y axis of occluded regions that belong to significant changes. Referring to equation (7.2), if the possibility percentage value ( $P$ ) of an occluded region below 49, it means that the occluded region might be too small and can be classified into an unimportant change.

The following Figs. depict results of performing the FIS1 to images in Figs. 7.1, 7.2 (a) – (b), 7.3 (a) – (c), 7.4 (a) – (c) and 7.5 (a) – (c). As seen in the following Figs., occluded regions, whose possibility percentage values are lower than 49, have been removed by the FIS1. Moreover, these removed regions often have smaller areas and random contours (i.e., they do not have specific shapes and could form any shape).



(a)

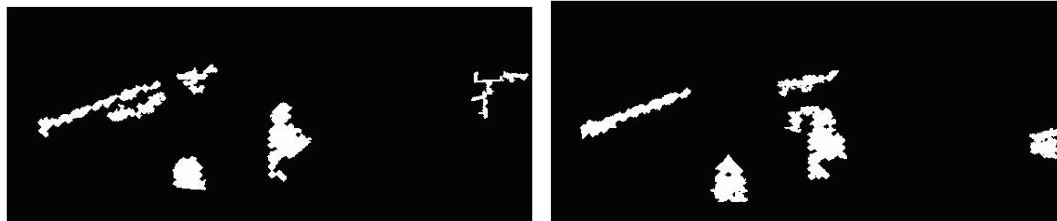


(b)



(c)

Fig. 7.7 The HDS1\_II-1 (a), HDS1\_II-5 (b) and HDS1\_II-9 (c)



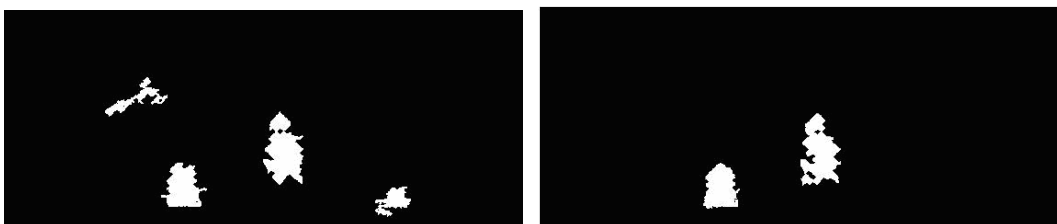
(a)

(b)



(c)

Fig. 7.8 The HDS1\_II-2 (a), HDS1\_II-6 (b) and HDS1\_II-10 (c)



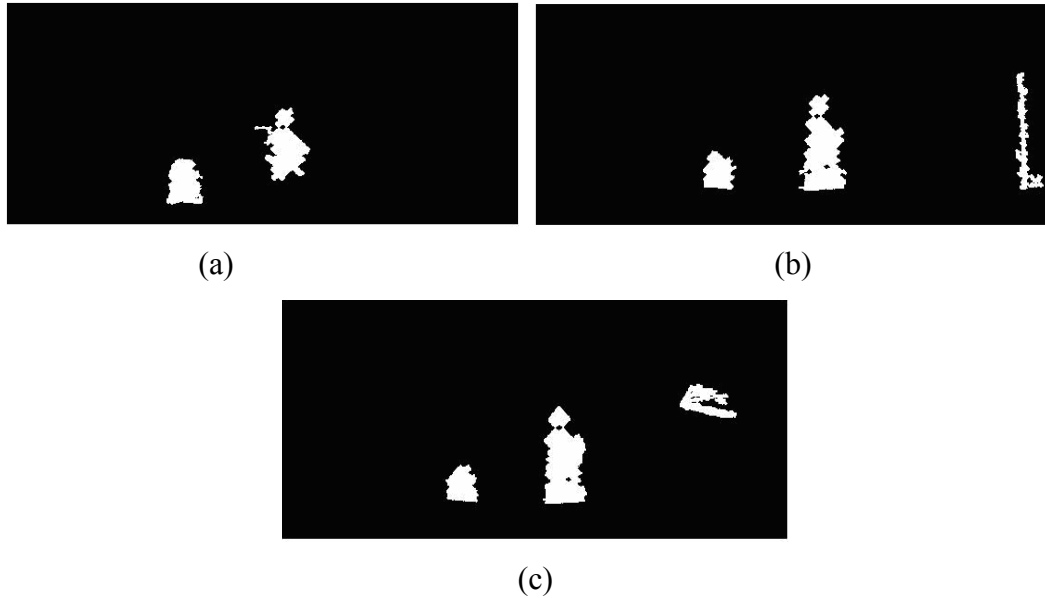
(a)

(b)



(c)

Fig. 7.9 The HDS1\_II-3 (a), HDS1\_II-7 (b) and HDS1\_II-11 (c)



**Fig. 7.10** The HDS1\_II-4 (a), HDS1\_II-8 (b) and HDS1\_II-12 (c)

As can be observed in all HDS1\_IIs above, both significant changes (i.e., a school bag and an intruder) are detected. However, unimportant changes (false positives) also appear in several HDS1\_IIs. To minimize these false positives, a template subtraction approach is developed and will be described in the next section.

## 7.4 Template Subtraction Approach

In reducing false positives (unimportant changes), a template subtraction approach is applied to the HDS1\_IIs. The template subtraction approach is outlined as follows.

For every object in the  $B_6(x, y)$  (i.e., a HDS1\_II, for example the HDS1\_II-1),

- 1.1 Extract bounding box parameters of the object including upper left corner of the object bounding box in X and Y axes, width and height of the object,
- 1.2 Based on these parameters, crop a template from its ROI2\_RI and a template from its ROI2\_II,
- 1.3 Subtract both templates by using the following equation,

$$S(x, y) = |T_{RI}(x, y) - T_{II}(x, y)| \quad (7.3)$$

where  $S(x, y)$  is the subtracted template,  $T_{RI}(x, y)$  is a template cropped from the ROI2\_RI and  $T_{II}(x, y)$  is a template cropped from the ROI2\_II,

- 1.4 Apply an automated global thresholding (Otsu, 1979) to the subtracted template in producing a binary image ( $B_7(x, y)$ ),
- 1.5 Decide whether or not there is a significant change in both templates by using the following equation,

$$B_8(x, y) = \begin{cases} B_7(x, y) & \text{if } (x, y) \geq \nu \\ 0 & \text{otherwise} \end{cases} \quad (7.4)$$

where  $\nu$  is a value set constant for any OMI at 8,000 pixels. The  $\nu$  value is experimentally determined in which 8,000 pixels represent a half size of a real school bag when an 8 MB image was captured by a digital camera from 6 meters distance from the scene.

- 1.6 Map the  $B_8(x, y)$  to the changed mask by using the following mapping equation,

$$X_{CM} = B_8(x) + Y_{BB} \quad (7.5)$$

$$Y_{CM} = B_8(y) + X_{BB} \quad (7.6)$$

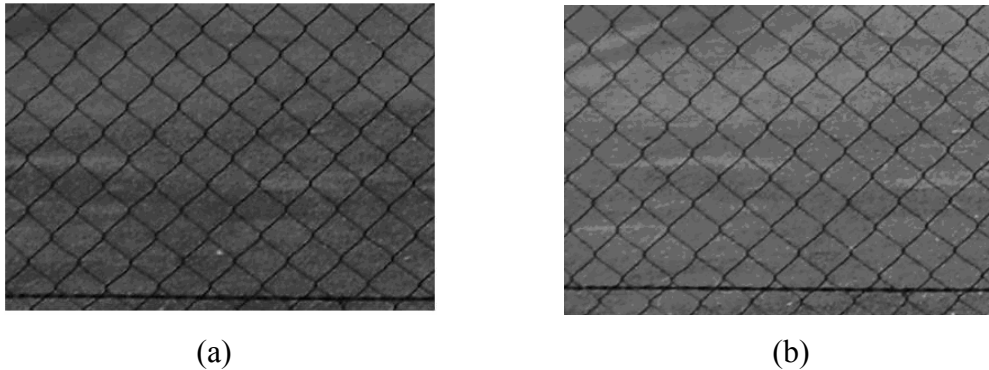
where  $X_{CM}$  and  $Y_{CM}$  are coordinate values of the changed mask in X and Y axes.  $B_8(x)$  and  $B_8(y)$  are coordinate values of the  $B_8(x, y)$  in X and Y axes.  $Y_{BB}$  and  $X_{BB}$  are upper left corner of the object bounding box in Y and X axes. The changed mask is  $B_9(X_{CM}, Y_{CM})$

- 1.7 Repeat steps 1.1 - 1.6 for other objects in the  $B_6(x, y)$ .

The following paragraphs describe results of performing the template subtraction approach to the HDS1\_II-1 (Fig. 7.7 (a)). According to the step 1.1 of the template subtraction approach, bounding box parameters of occluded regions



in the HDS1\_II-1 are extracted. Based on these parameters, templates are cropped from the ROI2\_RI-1 (Fig. 5.2 (b)) and ROI2\_II-1 (Fig. 5.2 (a)). Figs. 7.11 (a) and (b), below, depict two templates cropped from the ROI2\_RI-1 and ROI2\_II-1 based on information of bounding box parameters of an occluded region, which belongs to an unimportant change, in the HDS1\_II-1.



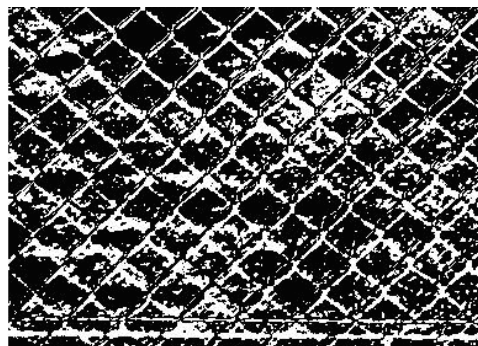
**Fig. 7.11** Two templates cropped from the ROI2\_RI-1 (a) and ROI2\_II-1 (b) based on information of bounding box parameters of an object in the HDS1\_II-1

As seen in Figs. 7.11 (a) and 7.11 (b), both templates have similar textures; however, illuminations are observably difference in both templates. Moreover, fence wires in both templates apparently have disparities especially horizontal wires.

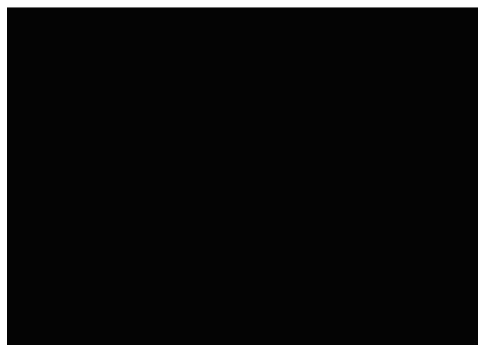
To check whether or not there is disappearing and/or appearing of a big object in both templates, both templates are subtracted. Fig. 7.12, below, depicts the subtracted image of both templates. As seen in Fig. 7.12, below, disparities apparently appear in vertical fence wires. Next, the subtracted image is converted into a binary image by applying an automated global thresholding proposed by Otsu (1979). The binary image is depicted in Fig. 7.13, below. Furthermore, big objects whose sizes at least 8,000 pixels (i.e., an object whose size is 15 cm by 15 cm in reality) are searched in the binary image. When a big object, whose size equal or greater than 8,000 pixels, is found in the binary image, then a true logic (logical 1) is produced to indicate that there is disappearing or appearing of an object in both templates. When there is no big object in the binary image, the object in the binary image is removed. Fig. 7.14, below, depicts the result of searching a big object whose size at least 8,000 pixels.



**Fig. 7.12** The subtracted image of both templates



**Fig. 7.13** The binary image after applying an automated global thresholding



**Fig. 7.14** The result of searching a big object in the binary image in Fig. 7.13. When an empty image is produced by the template subtraction approach, a false logic (logical 0) is produced to indicate that there is no disappearing and/or appearing of an object in both templates

Figs. 7.15 (a) and (b), below, depict other two templates cropped from the ROI2\_RI-1 and ROI2\_II-1 based on information of bounding box parameters of another occluded region in which a significant change appears in both templates.



(a)



(b)

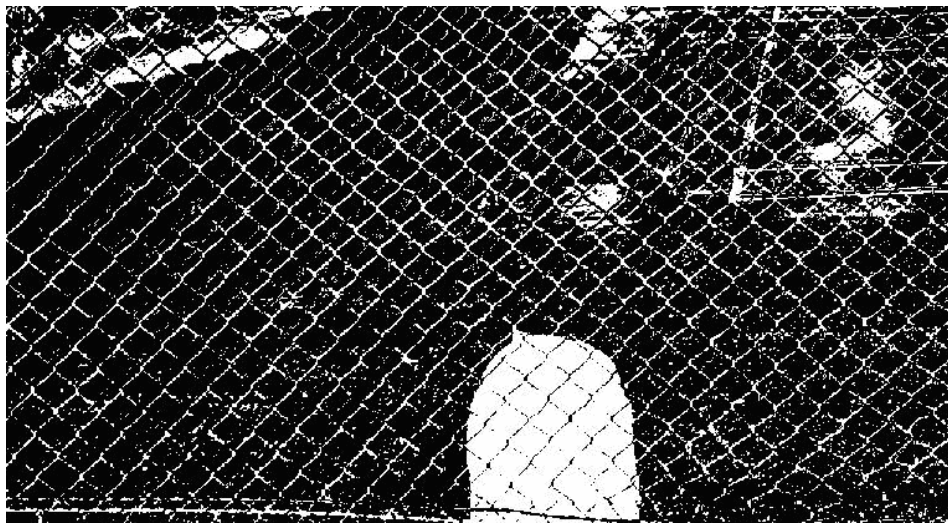
**Fig. 7.15** Two new templates cropped from the ROI2\_RI-1 (a) and ROI2\_II-1 (b) based on information of bounding box parameters of another object in the HDS1\_II-1

As seen in Figs. 7.15 (a) and (b), both templates have similar textures; however, there is disappearing of a school bag in both templates. To detect the disappearing a big object in both templates, both templates are subtracted and converted into a binary image by performing an automated global thresholding. Next, searching of big objects that have sizes equal or greater than 8,000 pixels is performing to the

binary image. The following Figs. depict results of detecting the disappearing of a big object in both templates.



**Fig. 7.16** The subtracted image between both templates in Figs. 7.15 (a) and (b)



**Fig. 7.17** The binary image produced after performing an automated global thresholding

When the template subtraction approach produces a big object after performing searching process (see Fig. 7.18, below), it means that there is disappearing of a big object in both templates. The searching process further produces a true logic to the template subtraction approach and the big object is included in the changed mask by applying a mapping process.



**Fig. 7.18** The result of searching a big object in the binary image

The following Figs. depict results of performing the template subtraction approach to other two templates cropped based on information of bounding box parameters of an occluded region that has presence of a new object in both templates.



(a)

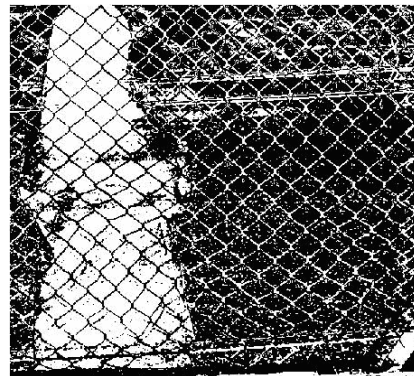


(b)

**Fig. 7.19** Two new templates cropped from the ROI2\_RI-1 (a) and ROI2\_II-1 (b)

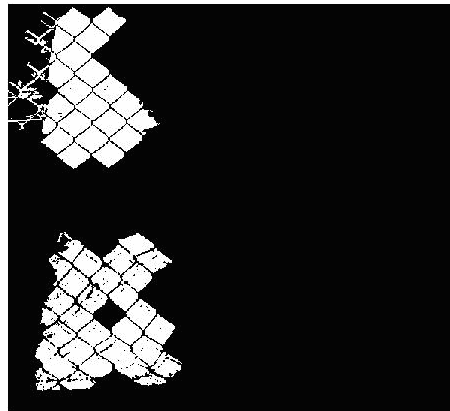


(a)



(b)

**Fig. 7.20** The subtracted image (a) of both templates in Figs. 7.19 (a) and (b), and the binary image (b) after performing an automated global thresholding

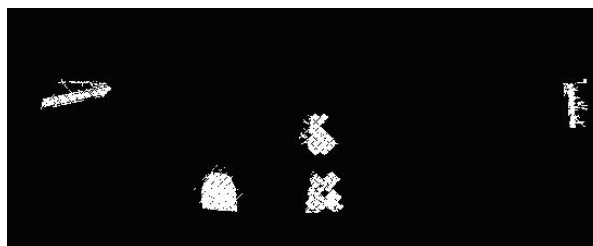


**Fig. 7.21** The result of searching big objects that have sizes equal or greater than 8,000 pixels in the binary image in Fig. 7.20 (b)

As seen in Fig. 7.20 (b), the intruder has been divided by edges of fence wires into several separated small objects. Referring to equation (7.4), separated small objects, whose sizes are less than 8,000 pixels, are removed from Fig. 7.20 (b) in generating an image in Fig. 7.21. As a result, separated small objects, whose sizes are equal or greater than 8,000 pixels, survive as shown in Fig. 7.21.

Next, objects in the Fig. 7.21 are mapped to the changed mask, referred to as the first changed mask of input image 1 (CHM1\_II-1).

The template subtraction approach is applied to all occluded regions in the HDS1\_II-1. Fig. 7.22, below, depicts the CHM1\_II-1.



**Fig. 7.22** The CHM1\_II-1 generated from the OMI\_II-1

As seen in Fig. 7.22, above, there are five objects. Three objects in the middle of the CHM1\_II-1 are significant changes (true positives) and other objects in the left and right sides of the CHM1\_II-1 are false positives. Changed masks of other OMI\_IIs will be presented in the next section.

### 7.5 Changed Masks

To obtain changed masks, the first hybrid decision-making system and template subtraction approach are applied to all OMI\_IIs. Figs. 7.23 (a) and (b) depict changed masks produced by the first algorithm from the ROI2\_II-5 and ROI2\_II-9.



Fig. 7.23 The CHM1\_II-5 (a) and CHM1\_II-9 (b)

As seen in Figs. 7.23 (a) and (b), objects appearing and disappearing behind fence wires observably appear in both changed masks. The school bag appears in its original contour and the intruder is divided into two separated small objects. Although the intruder is not in its original contour, the first algorithm can detect approximated locations of these significant changes.

Figs. 7.24 (a) – (c) depict changed masks of the ROI2\_II-2, ROI2\_II-6 and ROI2\_II-10.

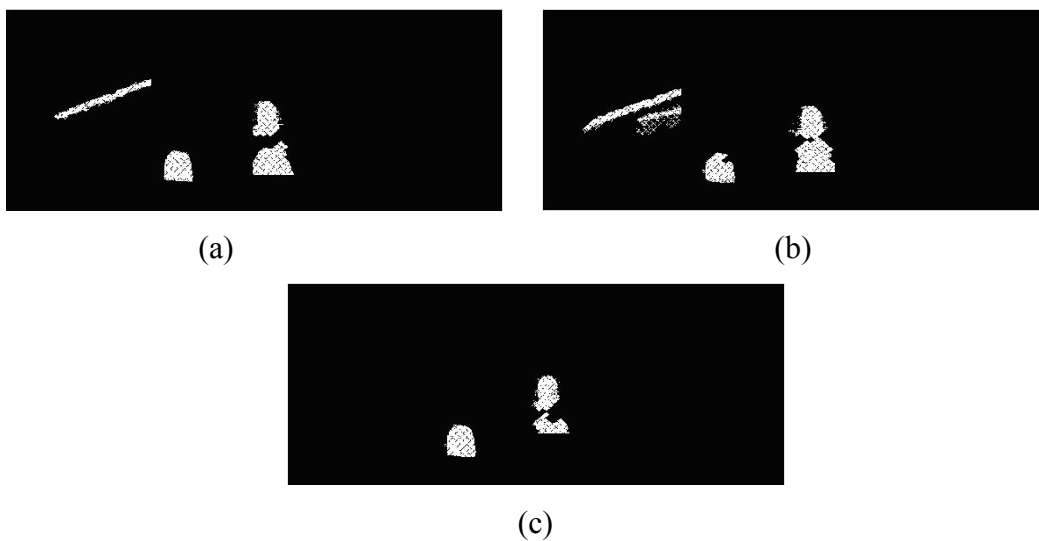
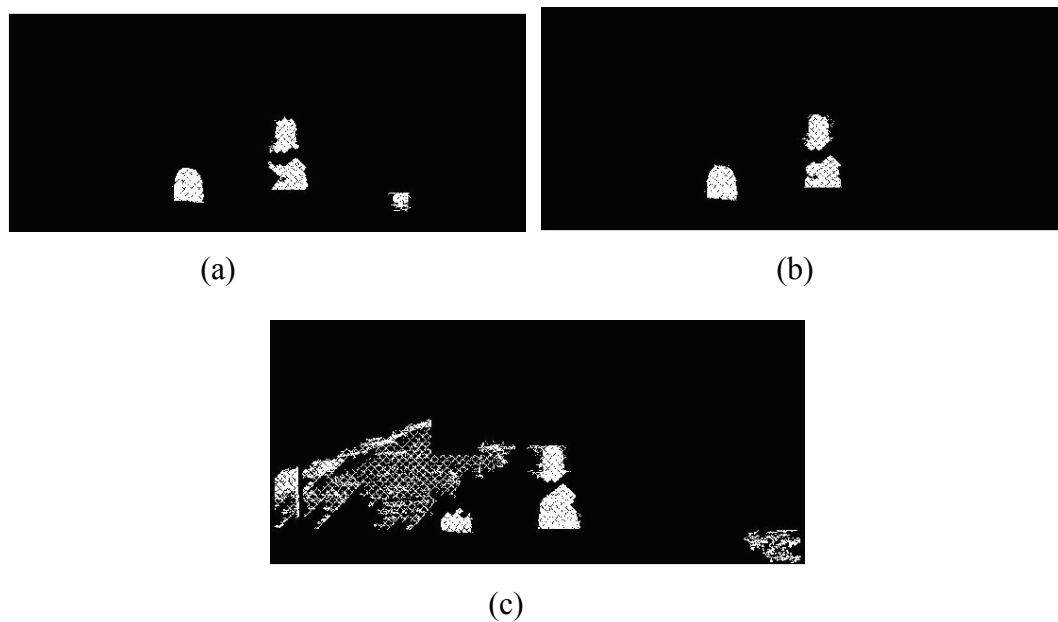


Fig. 7.24 The CHM1\_II-2 (a), CHM1\_II-6 (b) and CHM1\_II-9 (c)

As seen in Figs. 7.24 (a) – (c), there are a few false positives in the left top of the changed masks. These false positives are edges of fence wires and a surface of the wall. In contrast, there are no false positives in Fig. 7.24 (c) and the intruder is divided into two separated small objects as well.

Changed masks detected from the ROI2\_II-3, ROI2\_II-7 and ROI2\_II-11 by the first algorithm are depicted in Figs. 7.25 (a) – (c).

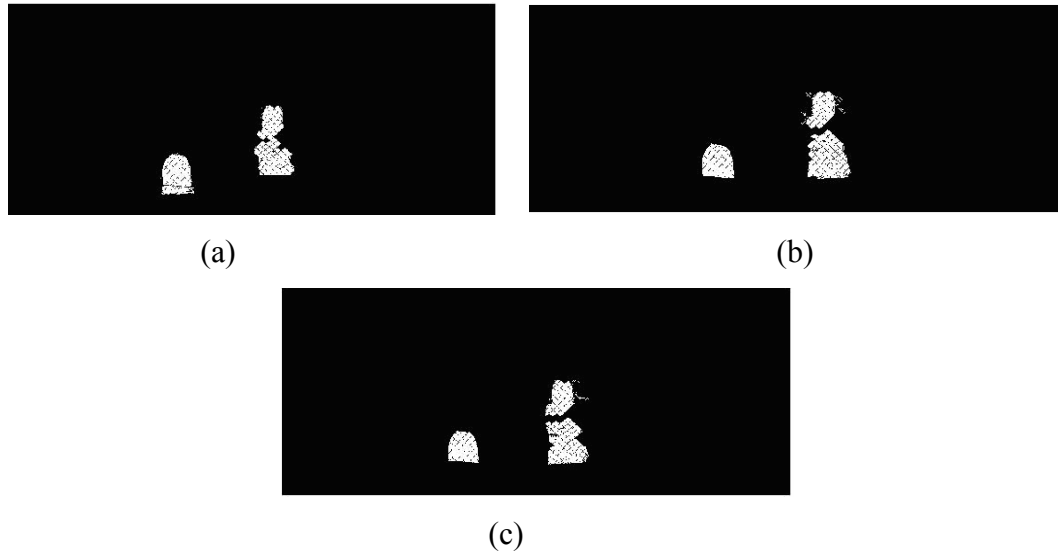


**Fig. 7.25** The CHM1\_II-3 (a), CHM1\_II-7 (b) and CHM1\_II-11 (c)

As can be seen in Fig. 7.25 (a), both significant changes are detected and an object attached in front of fence wires is detected as well. Moreover, there are no false positives in Fig. 7.25 (b). On the contrary, there are several false positives in Fig. 7.25 (c). The false positives are surface of the wall, edges of fence wires and the grass in front of fence wires. These kinds of false positives are very difficult to prevent because specular reflections that occur on fence wires, surface of the wall and the grass.

Fig. 7.26 (a) – (c) depict changed masks detected from the ROI2\_II-4, ROI2\_II-8 and ROI2\_II-12.





**Fig. 7.26** The CHM1\_II-4 (a), CHM1\_II-8 (b) and CHM1\_II-12 (c)

As seen in Figs. 7.26 (a) – (c), there are no false positives. Both the school bag and intruder are close to their original contours as shown in Fig. 7.26 (a). The intruder is separated into two objects in Figs. 7.26 (b) and 7.26 (c).

## 7.6 Concluding Remarks

The first algorithm presented in this chapter is used to detect only objects appearing and disappearing behind fence wires. Multiple outdoor images of the same scene containing fence wires captured by a mobile camera from slightly different positions, angles and at different times are used as inputs into the first algorithm. There are two significant changes (i.e., the school bag represents an object absence and the intruder denotes an object presence) in every input image. The first algorithm must be able to detect both significant changes in every input image.

The first algorithm is designed to detect regions of change (objects) whose sizes are from 8,000 pixels to 90,000 pixels. The minimum and maximum values represent the sizes of an object in the real life with the minimum dimension of 11 cm by 11 cm and the maximum dimension of 200 cm by 75 cm in height and width when an 8MB image captured from 600 cm distance from the digital camera to the scene.

Objects attached in front of fence wires, two small boxes, also emerge in every occlusion map image since they belong to occluded regions. However, they do not belong to significant changes since their sizes in pixels are not in the range of 8,000 - 90,000 pixels. As a consequence, they will be eliminated by the FIS1. If the FIS1 is designed to detect them as well; as a result, numerous unimportant changes, false positives, will emerge as well. The first algorithm may be able to detect objects in front of fence wires as long as their sizes in pixels are in the range of 8,000 - 90,000 pixels.

Changed masks produced by the first algorithm depict that both significant changes are detected in every input image. The intruder is often separated to two objects in the changed masks. Moreover, the school bag is often close to its original contour.

False positives observably appear in four changed masks (see Figs. 7.24 (a), 7.24 (b), 7.25 (a) and 7.25 (c)). These false positives are surface the wall, edges of fence wires and the grass in front of fence wires. Specular reflections that occur on surface of the wall, fence wires and the grass are mainly caused of these false positives since significant illumination variation happens during the day.

## CHAPTER 8

---

### DETECTING EDGES OF FENCE WIRES

In this chapter, a combination of the Sobel edge detector and an adaptive thresholding technique is presented in detecting edges of fence wires. This chapter begins with the background of extracting edges of fence wires. Next, an algorithm is overviewed. A section of experimental results and discussion is then presented and finally concluding remarks are drawn.

#### 8.1 Background

One of the main goals of this research is to detect breaches in the integrity of fence wires and attached objects in front of fence wires. Normally, intruders breach fence wires by making sizeable holes. The recognition of these types of breaches can be determined in an image by detecting the discontinuities in edges of fence wires. However, this will require an image processing system that can detect edges of fence wires in the first place so that a higher level automated analysis system can search for and detect such breaches.

Automated extraction of edges of outdoor fence wires from an image is in general a very difficult problem. Not only the wires are very thin, the problem is that there could be considerable background clutter, as well as significant daytime variation in illumination across fence wires.

Many edge detectors such as the Sobel, Prewitt, Roberts operators (Wang *et al.*, 2003), the Laplacian of Gaussian (LoG) operator (Basu, 2002) and the Canny operator (Peihua, 2007) were discussed in the literature (Kang and Wang, 2007). The edge detectors are generally divided into two categories: derivative-based and gradient-based. Derivative-based methods take first or second derivatives on each pixel in the image. In the case of the first derivative, if there is a rapid change of intensity at a pixel, then the pixel is categorized as an edge in the image. While in

the case of the second derivatives, if there is a zero pixel value at a pixel, referred to as zero-crossing, the pixel is classified as an edge in the image. As a consequence of taking the derivative of each pixel of the image, derivative-based methods require considerable computer memories and computational times; hence, they are not practical in real-time applications. To overcome this issue, an operation, known as kernel operation, is performed in such derivative-based methods.

The Sobel, Roberts and Prewitt operators belong to the first derivative-based method (Wang, *et al.*, 2003). The Sobel, Roberts and Prewitt operators find edges in an intensity image using approximations to the derivative. They return edges at those points where the gradient of the intensity image is maximum. A second derivative-based method is the Laplacian of Gaussian (LoG) method (Basu, 2002). The Laplacian of Gaussian (LoG) method finds edges by looking for zero crossing after filtering an intensity image with a Laplacian of Gaussian filter.

The Canny edge detection method is a gradient-based approach (Peihua, 2007). It is probably the most used edge detector in machine vision because it is generally superior to other edge detection methods in terms of detection, localization and a single response to a true edge. To extract edges in an image, the Canny detector firstly applies a Gaussian kernel in order to reduce false alarms (the number of no-change pixels incorrectly detected as change). For each pixel, it takes gradient components and computes gradient magnitudes and directions. Secondly, it performs non-maximal suppression in finding of true edge locations. The Canny detector uses two thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges

The available edge detectors are able to detect edges of fence wires in gray-level images, referred to as edged images, but their robustness is susceptible to noise, background clutter and the variation in illumination. This poses a problem if it is desired to binarise edged images (e.g., for further automated high level analysis in detection of any breaches in the wire fence integrity) because of a need to find an appropriate global threshold level.

An edged based image processing algorithm (i.e., an edge detector that combines the Sobel edge detector with an adaptive thresholding technique) that

overcomes the above mentioned problems of varying illumination on fence wires, background clutter and tiny sizes of fence wires, and eliminates the search for a global threshold is presented in the next section.

## 8.2 Algorithm Overview

Regions of interest of registered input images (ROI2\_IIs) discussed in Chapter 5 are used as inputs into the following edge detection algorithm. The algorithm is outlined as follows.

### 8.2.1 The Sobel Edge Detector

Prior to detect regions of change on fence wires, edges of fence wires must be detected first. To detect edges of fence wires, the Sobel edge detector was chosen in this research. The reason of choosing the Sobel detector is that it can be utilized to pursue edges in only diagonal directions since edges of fence wires used in this research are often composed in diagonal directions. Any edge detectors such as Prewitt, Roberts, Laplacian of Gaussian and Canny operators can be used to detect edges of fence wires.

To obtain an edged image,  $E(i, j)$ , from every ROI2\_II,  $I(i, j)$ , the Sobel detector is applied by referring to equations (8.1), (8.2), (8.3), (8.4), (8.5), (8.6) and (8.7) below [Baldock and Graham, 2000].

$$A_1 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \quad (8.1)$$

$$A_2 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix} \quad (8.2)$$

$$I_1(i, j) = I(i, j) * A_1 = \sum_{h=1}^m \sum_{k=1}^m A_1(h, k) I(i+h-1, j+k-1) \quad (8.3)$$

$$I_2(i, j) = I(i, j) * A_2 = \sum_{h=1}^m \sum_{k=1}^m A_2(h, k) I(i+h-1, j+k-1) \quad (8.4)$$

where \* indicates a discrete convolution,  $A_1$  and  $A_2$  are the Sobel masks, which are  $m \times m$  images and  $I(i, j)$  is a  $h \times w$  image.  $i$  runs from 1 to  $h - m + 1$  and  $j$  runs from 1 to  $w - m + 1$ .

Compute the gradient magnitude approximation at each pixel  $(i, j)$  as,

$$|G(i, j)| = |I_1(i, j)| + |I_2(i, j)| \quad (8.5)$$

To generate an edged image,  $E(i, j)$ , for every pixel in the  $G(i, j)$ ,

$$G(i, j) > 255 \rightarrow G(i, j) = 255 \quad (8.6)$$

$$E(i, j) = 255 - G(i, j) \quad (8.7)$$

As depicted in equation (8.7), edges of fence wires will be forced into darker areas and vice versa in the edged image.

### 8.2.2 An Adaptive Thresholding Technique

Output of the Sobel detector is an edged image which is normally in a gray level image. The edged image has often to be converted to a binary image for further process like boundary tracing. The available edge detectors use the global thresholding to convert the edged image to a binary image. As a result of non-uniform illumination that occurs on fence wires, the global thresholding may not be the best solution in converting the edged image to a binary image. To overcome non-uniform illumination that occurs on fence wires, an adaptive threshold technique is applied to the edged image in converting it to a binary image. The adaptive threshold technique is outlined by referring to equations (8.8), (8.9) and (8.10) below (Arques, et al., 2006; Tanjung, et. al., 2008).

Convolve the  $E(i, j)$  with the mean statistical operator,

$$\begin{aligned} C(i, j) &= E(i, j) * M_{fil}(h, k) \\ &= \sum_{h=1}^n \sum_{k=1}^n M_{fil}(h, k) E(i + h - 1, j + k - 1) \end{aligned} \quad (8.8)$$

where  $M_{fil}(h,k)$  is an image with a  $n$  by  $n$  filter containing equal weights whose values are ones, referred to as the averaging filter in which  $n$  is 50 and  $E(i, j)$  is a  $h$  x  $w$  image.  $i$  runs from 1 to  $h - n + 1$  and  $j$  runs from 1 to  $w - n + 1$ .

Subtract the  $E(i, j)$  from the convolved image,  $C(i, j)$ ,

$$S(i, j) = C(i, j) - E(i, j) \quad (8.9)$$

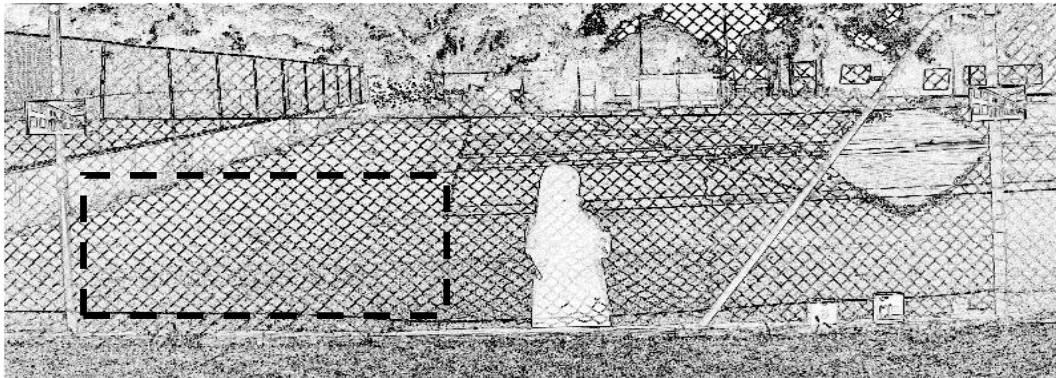
To produce a binary image,  $BW(x, y)$ , threshold the subtracted image,  $S(i, j)$ , with 0.

$$BW(x, y) = \begin{cases} 1 & \text{if } S(x, y) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (8.10)$$

As depicted in equation (8.10), zero is set constant as a threshold value by the adaptive thresholding technique in converting the edged image to a binary image although pixels values vary in the edged image as a result of illumination variation.

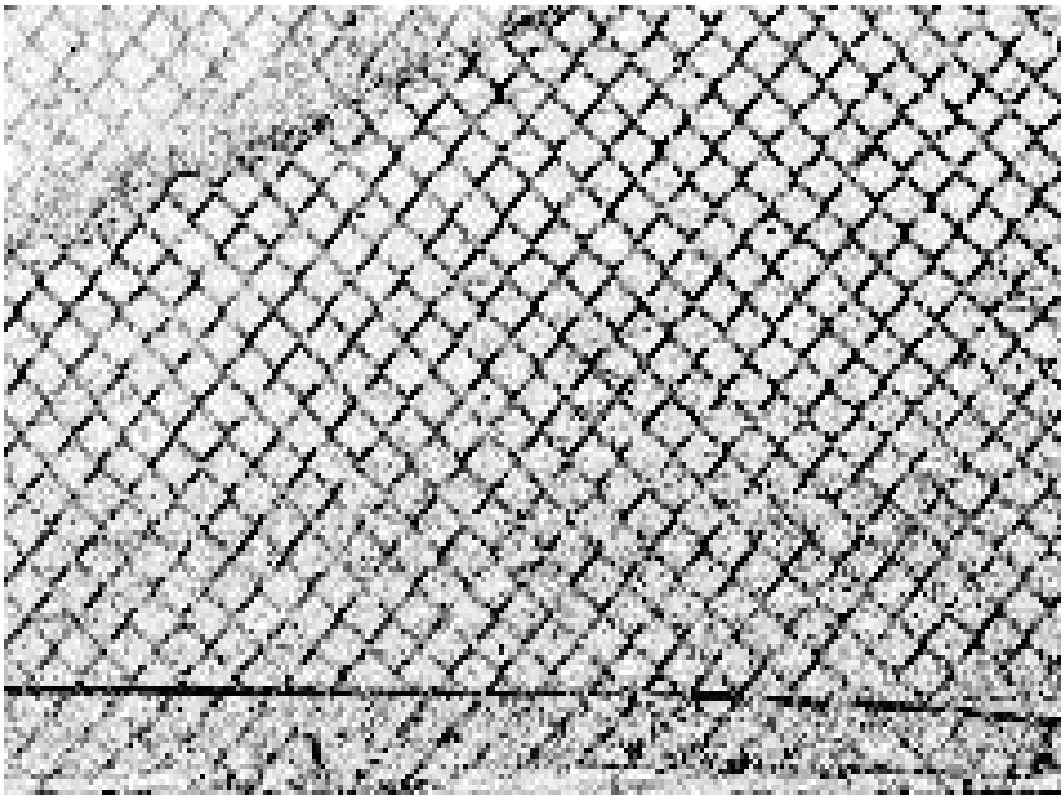
### 8.3 Experimental Results and Discussion

The following Figs. depict a result of applying the ROI2\_II-1 into the above algorithm.



**Fig. 8.1** The edged image,  $E_{II-1}(i, j)$ , produced after performing the Sobel edge detector into the ROI21\_II-1

The edged image,  $E_{II-1}(i, j)$ , in Fig. 8.1, above, is produced after applying equations 8.1, 8.2, 8.3, 8.4, 8.5, 8.6 and 8.7 into the ROI2\_II-1. As seen in Fig. 8.1, edges of fence wires are setup into darker pixels. Fig. 8.2, below, depicts the result of enlarging a small section at the bottom left of the edged image marked by a black thick dash line (see Fig. 8.1, above).

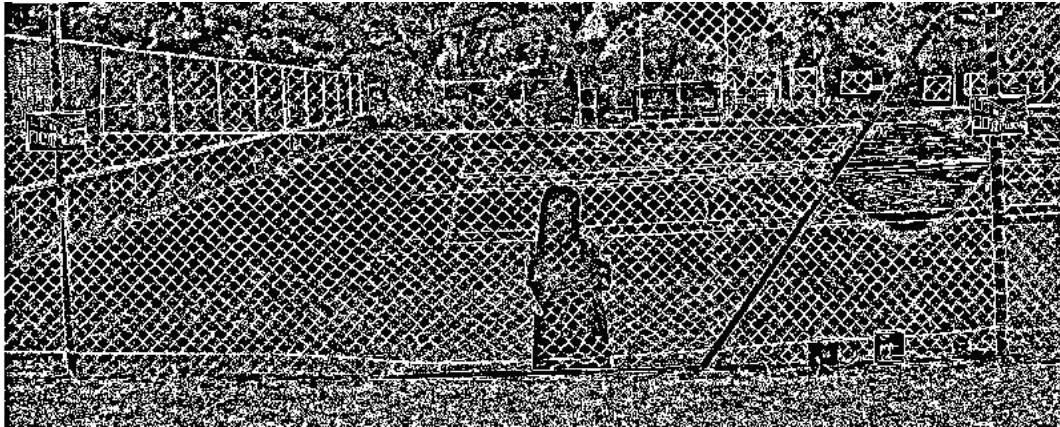


**Fig. 8.2** Enlarging the small left bottom part of the edged image

As seen in Fig. 8.2, above, edges of fence wires are not uniform in darker areas. The edges often change from darker areas into lighter areas. These conditions are caused of the non-uniform illumination that occurs across fence wires.

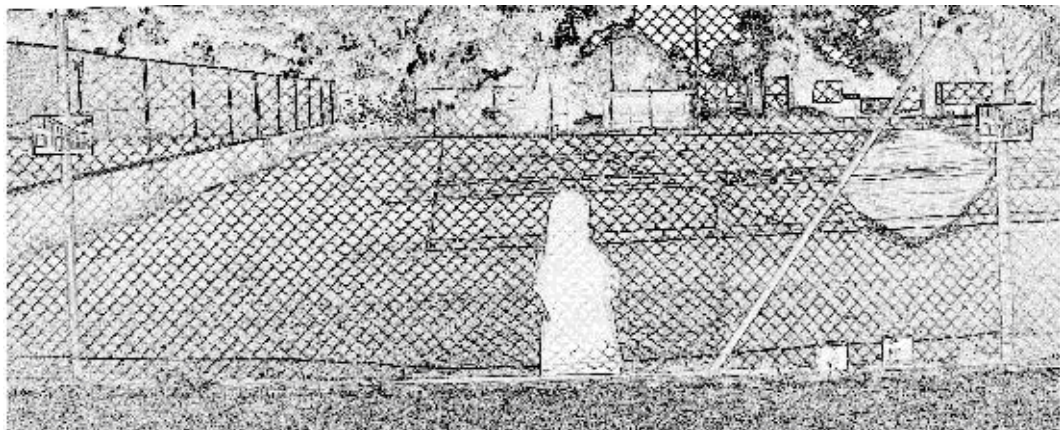
Referring to equations 8.8, 8.9 and 8.10, a binary image,  $BW_{II-1}(x, y)$ , was generated. Fig. 8.3, below, depicts the binary image after performing the adaptive thresholding towards the edged image in Fig. 8.1, above. As seen in Fig. 8.3, below, edges of fence wires are in lighter areas. Edges of fence wires visually appear in the binary image.



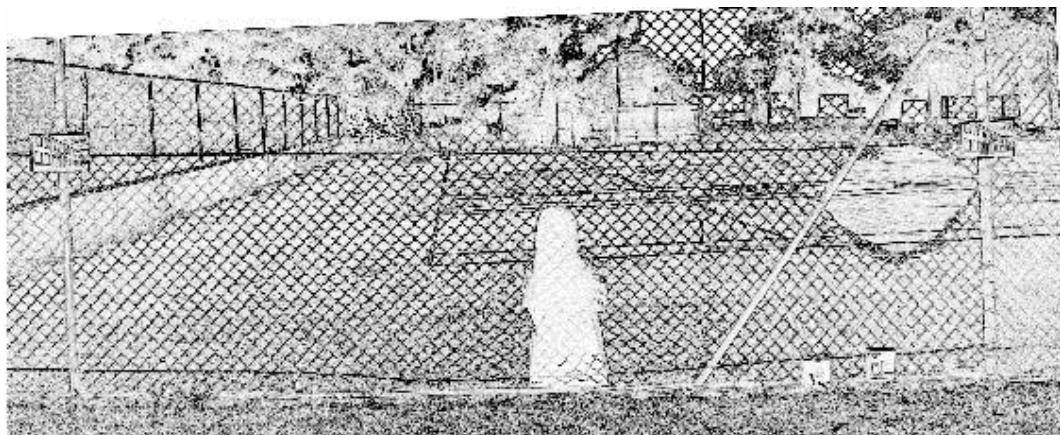


**Fig. 8.3** The binary image,  $(BW_{II-1}(x, y))$ , generated after performing the adaptive thresholding into the  $E_{II-1}(i, j)$

The following Figs. are edged images ( $E_{IIs}(i, j)$ ) of other ROI2\_IIs.

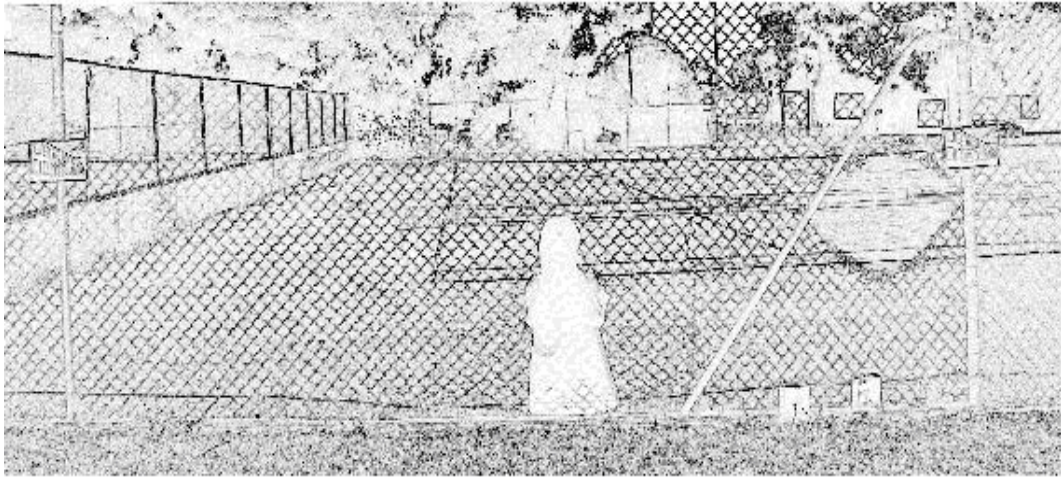


(a)

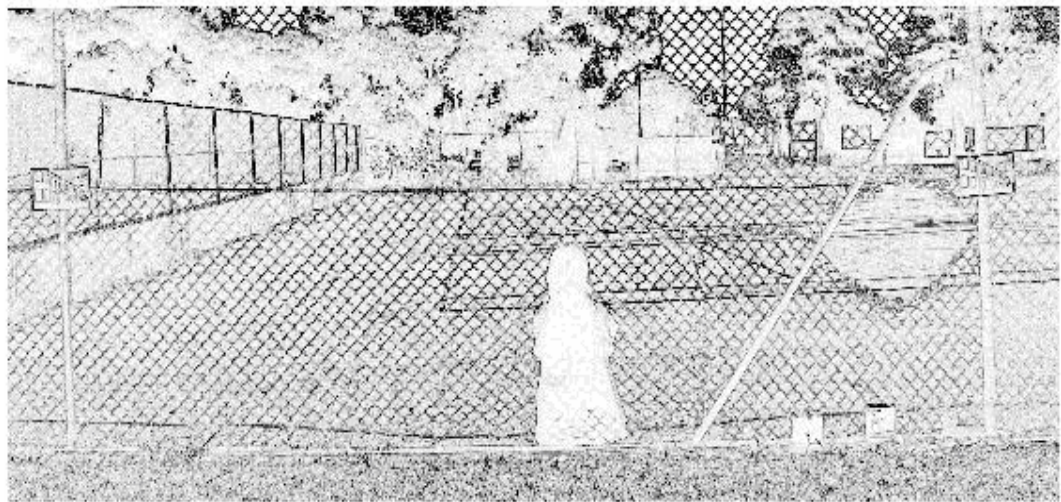


(b)

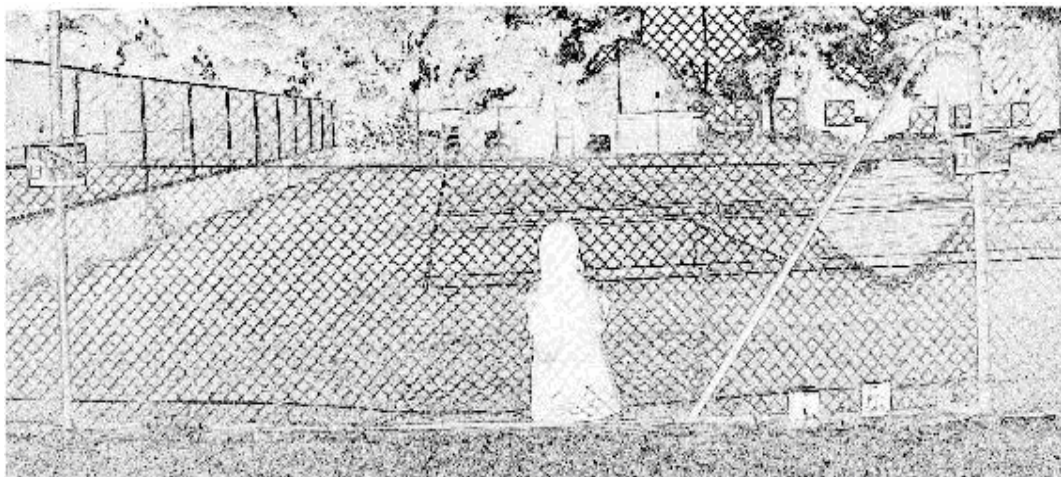
**Fig. 8.4** Edged images of the ROI2\_II-5 (a) and ROI2\_II-9 (b)



(a)

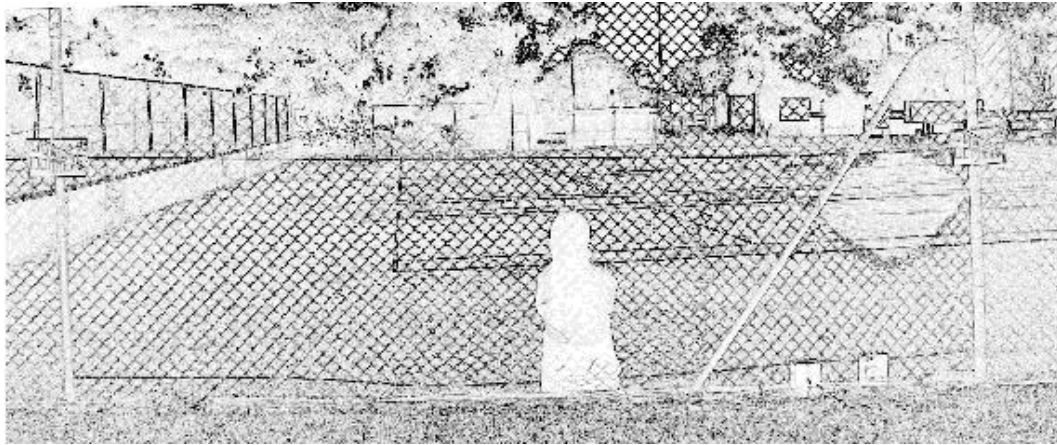


(b)

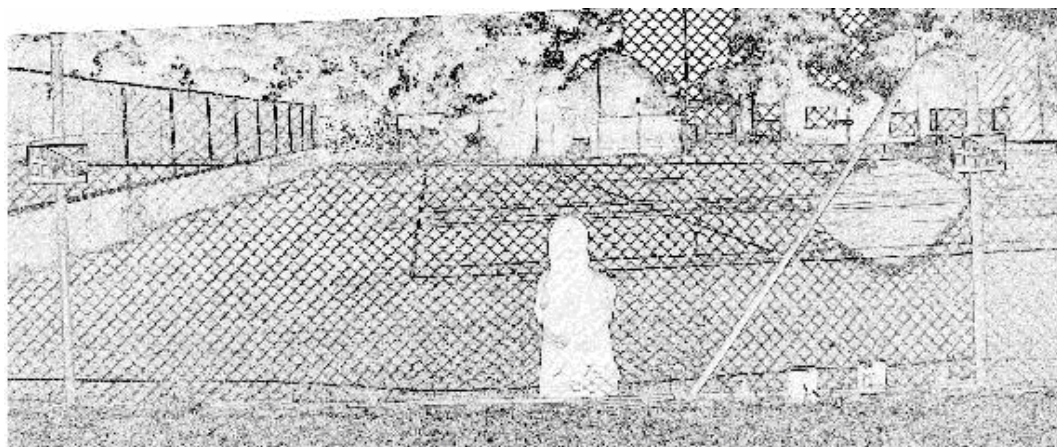


(c)

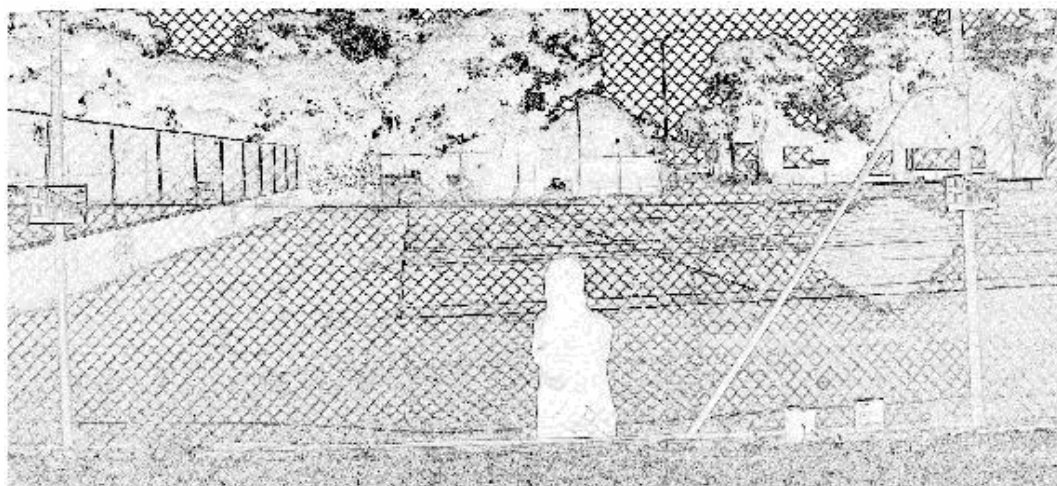
**Fig. 8.5** Edged images of the ROI2\_II-2 (a), ROI2\_II-6 (b) and ROI2\_II-10 (c)



(a)

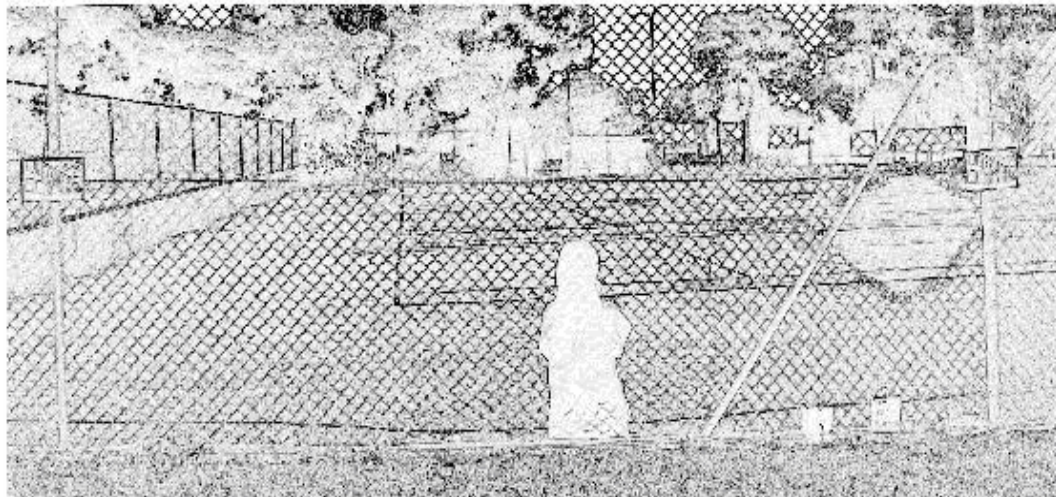


(b)

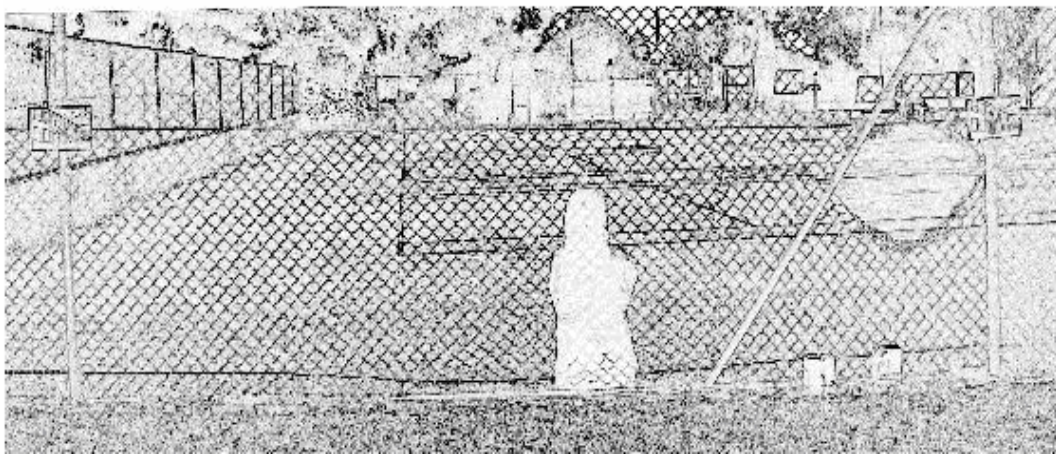


(c)

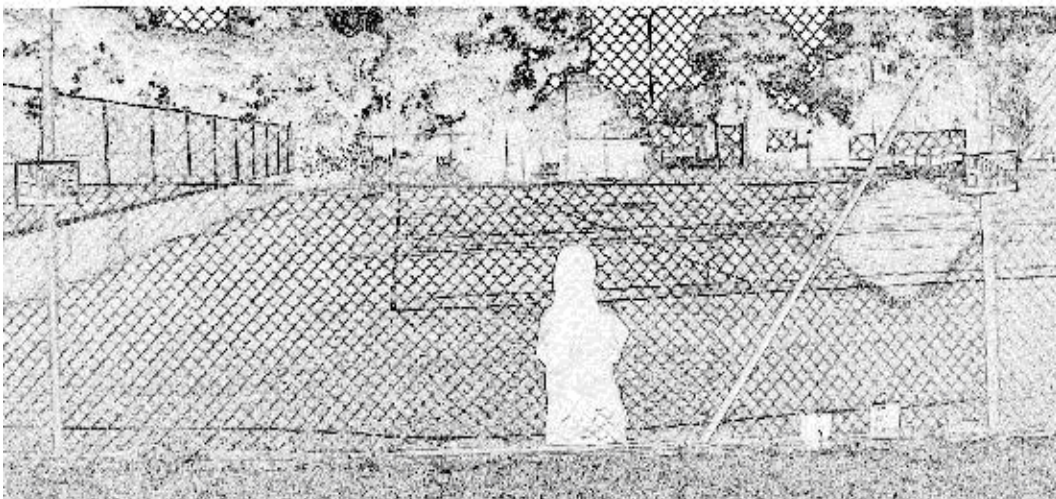
**Fig. 8.6** Edged images detected from the ROI2\_II-3 (a), ROI2\_II-7 (b) and ROI2\_II-11 (c)



(a)



(b)

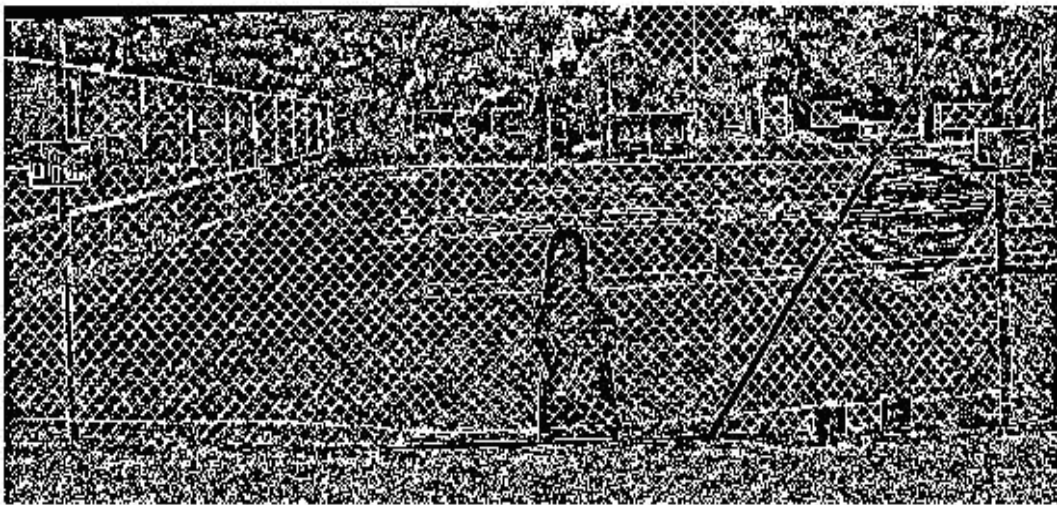


(c)

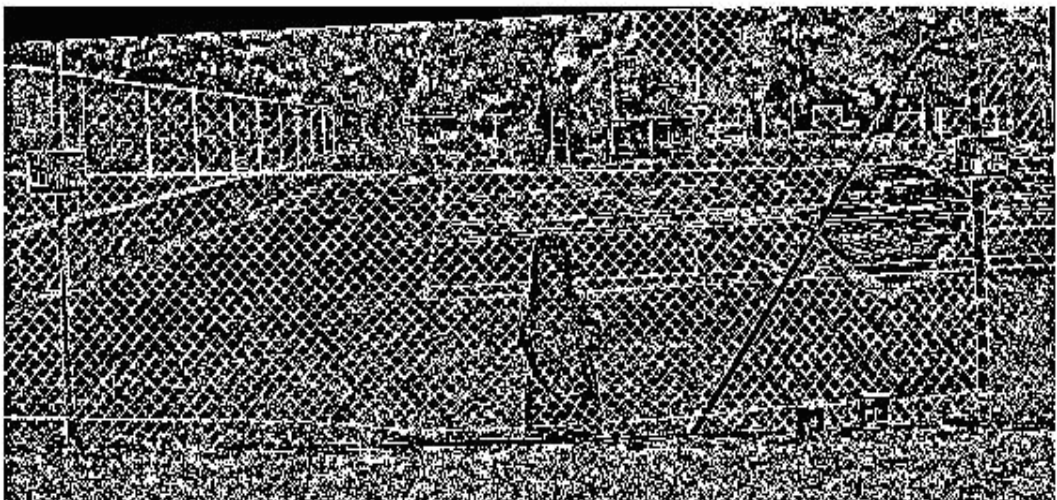
**Fig. 8.7** Edged images extracted from the ROI2\_II-4 (a), ROI2\_II-8 (b) and ROI2\_II-12 (c)

As seen in Figs. 8.4, 8.5, 8.6 and 8.7, above, edges of fence wires are in darker pixels. As a result of non-uniform illumination that occurs across fence wires, edges of fence wires often change to lighter pixels. Trees and buildings also absorb edges of fence wires and as a result, they vanish.

The next challenge is to convert edged images into binary images in which edges of fence wires have to appear in binary images. An adaptive thresholding technique is then applied on edged images. The following Figs. depict binary images produced after performing an adaptive thresholding on edged images.

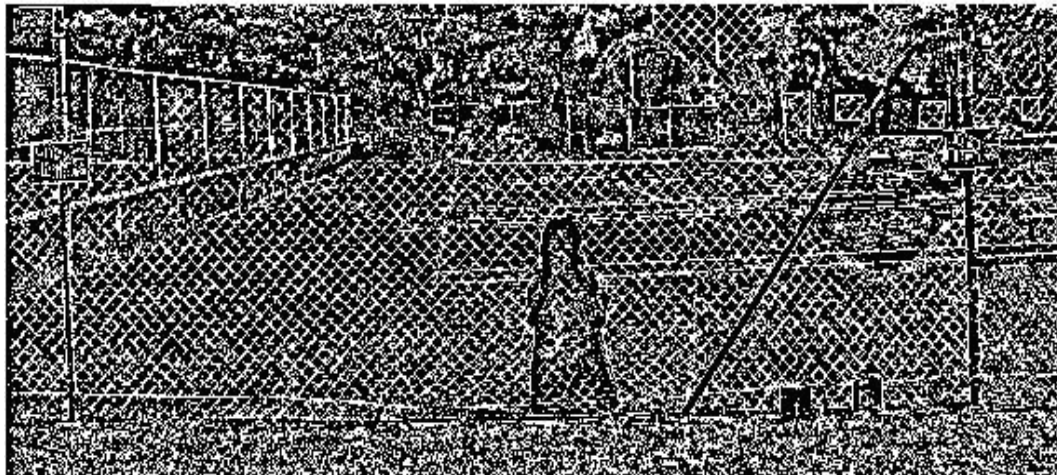


(a)

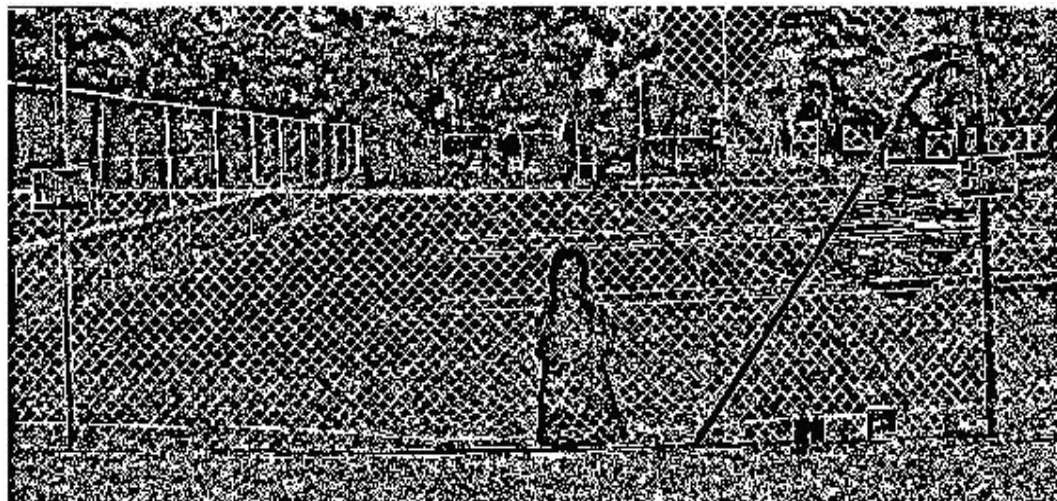


(b)

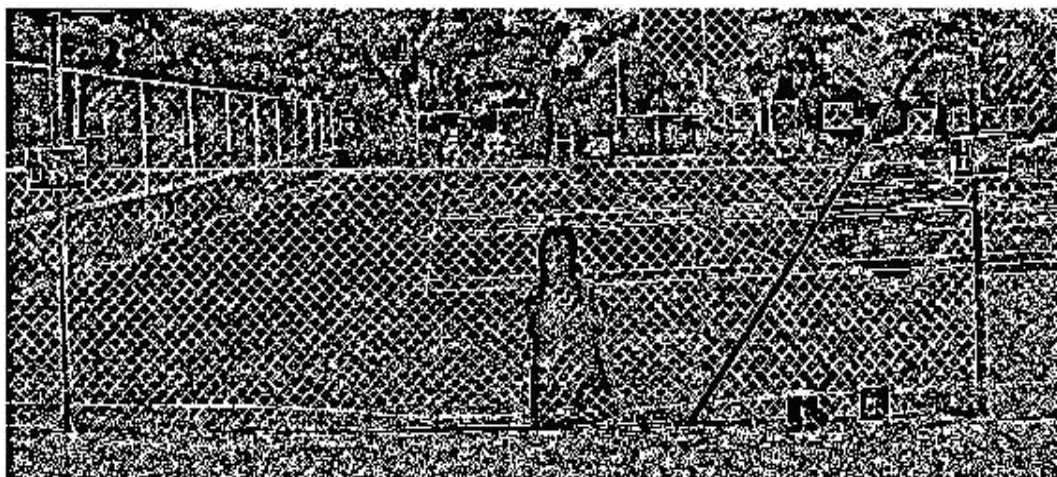
**Fig. 8.8** Binary images extracted from the  $E_{II-5}(i, j)$  (a) and  $E_{II-9}(I, j)$  (b) after applying the adaptive thresholding technique



(a)

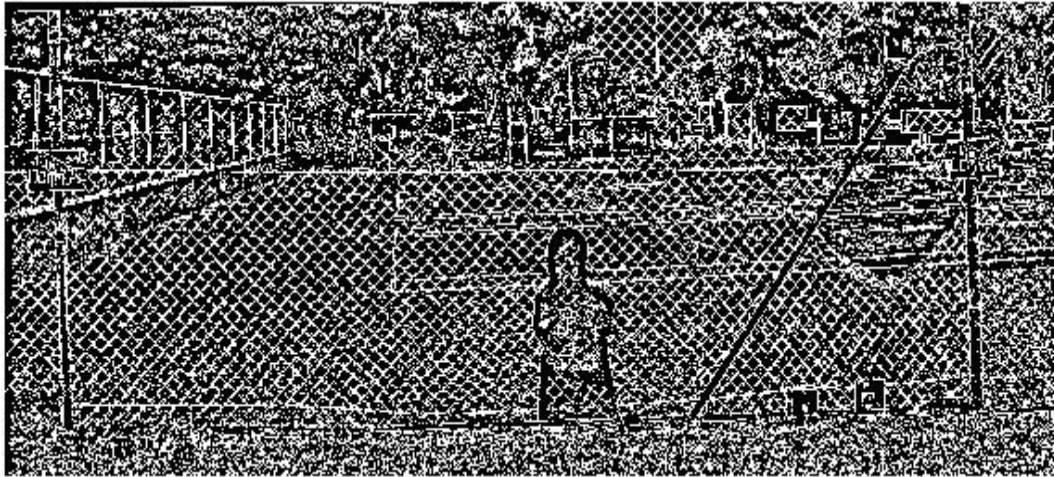


(b)



(c)

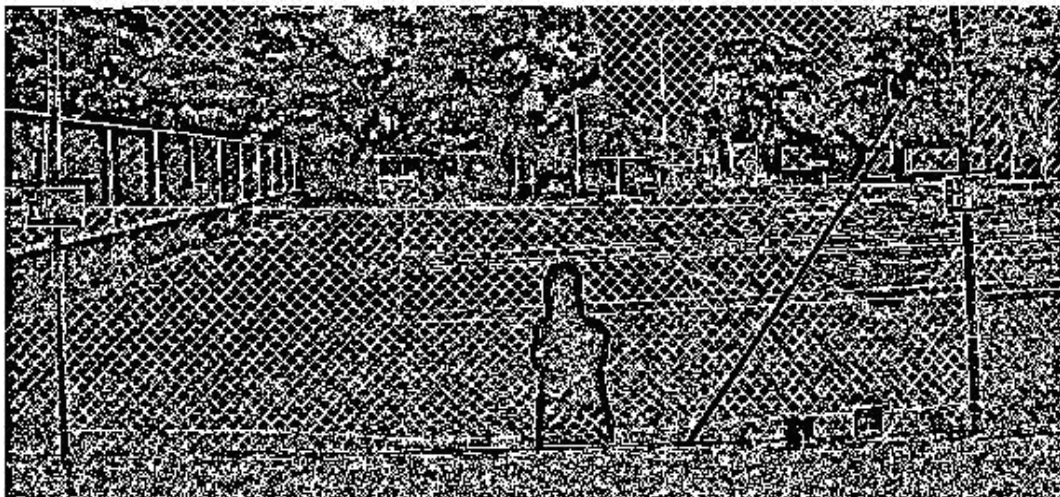
**Fig. 8.9** Binary images (i.e.,  $BW_{II-2}(x, y)$ ,  $BW_{II-6}(x, y)$  and  $BW_{II-10}(x, y)$ ) detected from the  $E_{II-2}(i, j)$  (a),  $E_{II-6}(i, j)$  (b) and  $E_{II-10}(i, j)$  (c)



(a)

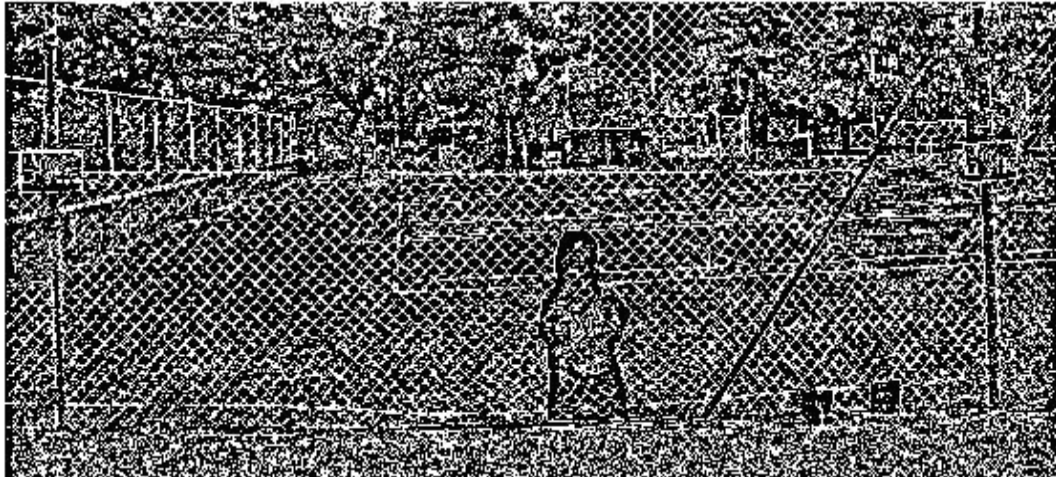


(b)

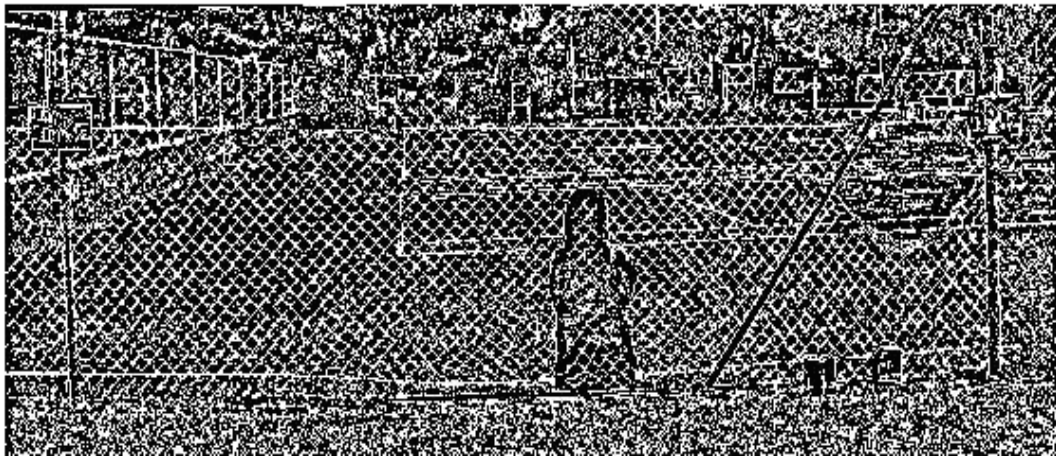


(c)

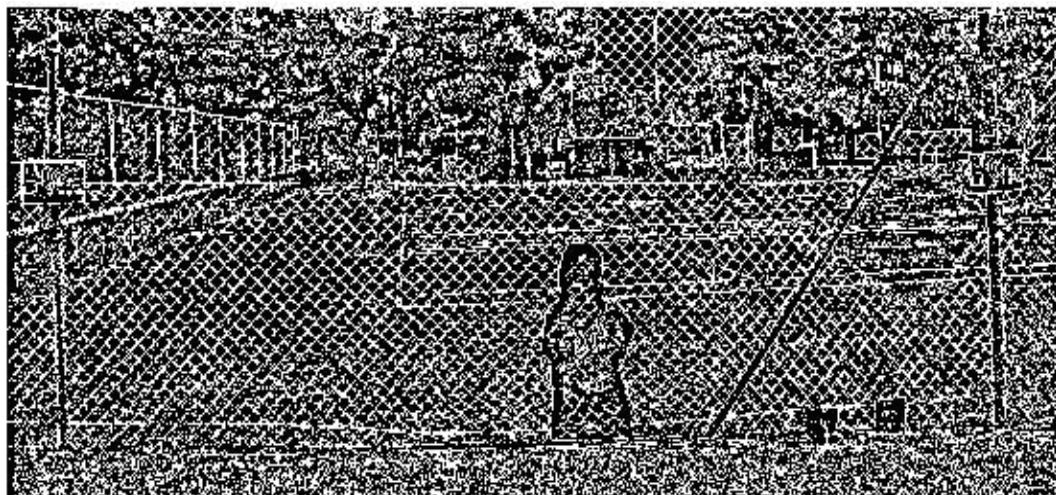
**Fig. 8.10** Binary images generated from the  $E_{II-3}(i, j)$  (a),  $E_{II-7}(i, j)$  (b) and  $E_{II-11}(i, j)$  (c) after performing the adaptive thresholding



(a)



(b)



(c)

**Fig. 8.11** Binary images (i.e.,  $BW_{II-4}(x, y)$ ,  $BW_{II-8}(x, y)$  and  $BW_{II-12}(x, y)$ ) extracted from the  $E_{II-4}(i, j)$  (a),  $E_{II-8}(i, j)$  (b) and  $E_{II-12}(i, j)$  (c)



As seen in Figs. 8.8, 8.9, 8.10 and 8.11, above, edges of fence wires are in lighter pixels. Moreover, they visually appear in each binary image.

Available edge detectors such as the Prewitt, Roberts, Laplacian of Gaussian and Canny operators can be used to generate edge images. For further processing like boundary tracing, edges images often have to be converted into binary images. The available edge detectors commonly use one global threshold value in converting edged images to binary images. The Canny operator uses two threshold values to generate binary images from edge images. In the case of detecting edges of fence wires during the day, the use of global threshold values is hard to be the best solution.

In fact, illumination varies during the day and fence wires are often built from metal material. As a consequence, specular reflections observably appear across fence wires during a sunny day and fence wires appear darker during an overcast day. As a result, non-uniform illumination occurs across fence wires. In such a situation, the process of fine tuning one or two appropriate global threshold values can become a complex and time-consuming task.

To overcome this complexity, the use of the adaptive thresholding can be the best solution. By using the adaptive thresholding, we do not need to search any threshold value since a zero threshold value can be setup constant to any edged images.

Experimental results in Figs. 8.8, 8.9, 8.10 and 8.11, above, show that edges of fence wires are accurately detected in all binary images. This demonstrates that the use of the Sobel operator combined with an adaptive thresholding technique is quite robust towards illumination variation as well as no manual parameter setting (i.e., global threshold values) is required for being connected into binary images.

## 8.4 Concluding Remarks

In the case of detecting edges of outdoor fence wires during the day, available edge detectors, which use one or two global threshold values in order to convert binary images from edged images, may be not the best solution because of non-uniform illumination that occurs across fence wires. To overcome the non-uniform illumination on fence wires, the use of the Sobel edge detector combined

with an adaptive threshold technique has been introduced and developed in this research. The Sobel operator is used to generate edged images from RO2\_IIs and the adaptive thresholding technique is utilized in converting edged images into binary images for further processing.

Advantages of the Sobel detector blended with the adaptive threshold technique are summarized as follows: (1) it is quite robust towards varying illumination conditions and (2) it is setup by using only a zero threshold value for all edged images (i.e., no manual parameter setting). In addition, the presented edge detector approach has been presented in the 2008 Australasian Conference on Robotics and Automation (ACRA 2008) (Tanjung *et al.*, 2008).

Edges of fence wires in binary images generated by the mixed Sobel edge detector and the adaptive thresholding can also be enhanced by performing morphological operations such as removing small objects and dilation by using a diagonal line direction structure element [Gonzalez and Woods, 2008]. When edges of fence wires can be detected without confusion from input images, detecting breaches on outdoor wire fences during the day can be achieved.

## CHAPTER 9

---

# DETECTING BREACHES IN THE INTEGRITY OF AND ATTACHED OBJECTS IN FRONT OF FENCE WIRES

Chapter 9 describes the second algorithm of the complete change detection method. The second algorithm is only used to detect breaches in the integrity of and attached objects in front of fence wires. The second algorithm is developed based on information of Chapter 8. Section 9.1 begins with problems that arise from these kinds of multiple outdoor images of the same scene containing fence wires. A detailed explanation of the second algorithm is presented in Section 9.2. Section 9.3 presents experimental results. Section 9.4 presents changed masks produced by the second algorithm. Concluding remarks are presented in Section 9.5.

### 9.1 Background

Detecting breaches in the integrity of fence wires and attached objects in front of fence wires while rejecting unimportant changes caused by camera motion, sensor noise, illumination variation and non-uniform attenuation in multiple outdoor images of the same scene containing fence wires captured by a mobile camera from slightly different positions, angles and at different times is a complex task. Complexity further increases when the change detection is performed in an outdoor scene (i.e., a lot of background clutter and a significant day-time variation in illumination) that contains fence wires. Outdoor fence wires are often built from thin metal substance (i.e., the wire width size of the fence used in this research is 2.5 mm). Specular reflections apparently appear on fence wires during the sunny day and fence wires seem darker during the overcast day. Consequently, non-uniform illumination observably occurs across fence wires.

There are two main problems addressed in this study. The first problem is how to extract edges of fence wires without missing considerable edges of fence wires in such complex scene prior to detect breaches in the integrity of fence wires. An investigation of edge detectors that are quite robust towards illumination variations is conducted in this study.

The second problem is how to detect breaches in the integrity of and attached objects in front of fence wires in binary images produced by the proposed edge detector. Any significant changes must be detected and located while any unimportant changes must be rejected by the automated change detection method.

## **9.2 Algorithm Overview**

The second algorithm consists of four main stages: (1) detecting edges of fence wires, (2) enhancing edges of fence wires, (3) removing diamond-shape areas and (4) determining significant or unimportant changes using the second hybrid decision-making system. Detailed explanations of the main stages are described in the following sub-sections.

### **9.2.1 Detecting Edges of Fence Wires**

Prior to detect regions of change of fence wires, edges of fence wires must be detected first. Detecting edges of fence wires in these kinds of multiple outdoor images is a very difficult problem. Not only the wires are very thin (i.e., 2.5 mm), the problem is that there could be considerable background clutter, a significant daytime variation in illumination across the fence, as well as parallax caused by camera motion. Since focus of the second algorithm is to detect only regions of change on particularly fence wires including breaches in the integrity of and attached objects in front of fence wires, parallax can be explicitly eliminated by detecting straightforward edges of fence wires.

The Sobel edge detector is used to produce a edged image from every ROI2\_II and in converting the edged image to a binary image for further process, an adaptive thresholding technique is applied to the edged image. An explanation of detecting edges of fence wires in the multiple outdoor images is described in Chapter 8.

### 9.2.2 Enhancing Edges of Fence Wires

Edges of fence wires in the binary image,  $BW(x, y)$ , produced by the combined Sobel detector and adaptive threshold have to enhance in order to provide enough information for further process. To enhance edges of fence wires, morphological operations such as dilation and erosion are used. First, small objects in the  $BW(x, y)$  are removed by referring to equation (9.1) below ((Gonzalez, R. C. & Woods, R. E., 2008).

$$BW_1(x, y) = \begin{cases} BW(x, y), & \text{if } (x, y) > 50 \text{ pixels} \\ 0 & , \text{otherwise} \end{cases} \quad (9.1)$$

where 50 pixels represent an object with the area of 100 square millimetres (the size of a small electronic resistor) when an image is captured from a 600 cm distance in the real life setting. It means that objects whose sizes are 50 pixels are categorized as unimportant changes in this experiment.

Next, dilation is applied to the  $BW_1(x, y)$  in which line structuring elements (i.e., the thickness of the line is 3 pixels) in 45 degree and 135 degree directions,  $B_{45}$  and  $B_{135}$  respectively, are used.

$$\begin{aligned} BW_2(x, y) &= BW_1(x, y) \oplus B_{45} \\ &= \bigcup \{B_{45} + a : a \in BW_1(x, y)\} \end{aligned} \quad (9.2)$$

$$\begin{aligned} BW_3(x, y) &= BW_2(x, y) \oplus B_{135} \\ &= \bigcup \{B_{135} + a : a \in BW_2(x, y)\} \end{aligned} \quad (9.3)$$

where  $a$  is a pixel value of the  $BW_1(x, y)$  or  $BW_2(x, y)$ .

As a result of dilation, edges of fence wires in the  $BW_3(x, y)$  become more thick than their original size in the  $BW_1(x, y)$ . The  $BW_3(x, y)$  is a new binary image.

### 9.2.3 Removing Diamond-Shape Areas

To detect breaches in the integrity of and attached objects in front of fence wires, an area-based algorithm is developed in the second algorithm. Since a chain-link mesh fence (i.e., a diamond netting fence) is used in this research, potential breaches and

attached objects will have different areas and shapes with diamond-shape objects in between edges of fence wires. By knowing an average area of these diamond-shape objects, diamond-shape objects can be removed based on information of the average area. The area-based algorithm will overcome background clutter since edges of fence wires divide big objects on the background into diamond-shape objects as well.

The area-based algorithm is outlined as follows. Firstly, each element of the  $BW_3(x,y)$  is complemented in generating another new binary image,  $CM_1(x,y)$  referring to equation (9.4) below.

$$CM_1(x,y) = \sim BW_3(x,y) \quad (9.4)$$

where  $\sim$  is a not logical operator.

Secondly, diamond-shape objects are removed from the  $CM_1(x,y)$  by referring to equation (9.5) below.

$$CM_2(x,y) = \begin{cases} CM_1(x,y) & \text{if } CM_1(x,y) > avgV \\ 0 & \text{otherwise} \end{cases} \quad (9.5)$$

The value of  $avgV$  was experimentally determined based on information of Table 9.1 below.

Next, any line-shape objects in the  $CM_2(x,y)$  is removed by referring to equation (9.6) below.

$$CM_3(x,y) = \begin{cases} CM_2(x,y) & \text{if } CM_2(x,y) < ecc \\ 0 & \text{otherwise} \end{cases} \quad (9.6)$$

where  $ecc$  is an eccentricity (i.e., the ratio of the distance between the foci of the ellipse and its major axis length) value. The  $ecc$  value is between 0 and 1. An ellipse whose eccentricity is 0 is actually a circle, while an ellipse whose eccentricity is 1 is a line segment. In this study, the  $ecc$  value was setup constant for every input image at 0.99.

**Table 9.1** Summarization of  $avgV$  values.  $d$  denotes the distance between a mobile-camera and fence wires in meter,  $s$  stands for a file size in Megabytes,  $w$  is the width of fence-wires in pixels and  $avgV$  was calculated by multiplying  $w$  with 1000 pixels. In this study, the file size of each input image was 8 MB and  $d$  was 6 m. Hence, the  $avgV$  value was setup constant for any input image at 3000 pixels.

$d$ (m)	$s$ (MB)	$w$ (pixels)	$avgV$ ( $w \times 1000$ pixels)
6	4	1	1000
5	4	2	2000
4	4	3	3000
3	4	4	4000
6	8	3	3000
5	8	4	4000
4	8	5	5000
3	8	6	6000

Finally, dilation is perform towards the  $CM_3(x, y)$  by referring to equation (9.7) (Gonzalez, R. C. & Woods, R. E., 2008).

$$CM_4(x, y) = CM_3(x, y) \oplus B_{rect} \quad (9.7)$$

where  $B_{rect}$  is a rectangle-shape structuring element whose size is 10 rows and 15 columns in the structuring element neighbourhood.

#### 9.2.4 The Second Hybrid Decision-Making System

Since the second algorithm is quite different with the first algorithm in terms of detecting regions of change in these kinds of multiple outdoor images, output of the second algorithm results will be quite different with output produced by the first algorithm in terms of object's areas and contours. Thus, a new decision-making system, referred to as the second hybrid decision-making system (HDS2), is developed in the second algorithm. The HDS2 is used to determine which objects in the  $CM_4(x, y)$  belong to significant or unimportant changes. The HDS2 consists of two sub-systems: crisp and fuzzy inference sub-systems. The followings are detailed explanations of crisp and fuzzy inference sub-systems.

**9.2.4.1 Second Crisp Inference Sub-System**

The second crisp inference system is utilized to remove objects in the  $CM_4(x, y)$  based on their positions in Y axis. Since searching of significant changes is limited from the ground to 1.5 meters height, objects positioned higher than 1.5 meters will be removed from the  $CM_4(x, y)$  by the second crisp inference system. Objects in 1.5 meters height represent the sky, tops of left and right posts of the wire fence and tops of trees and buildings. Equation (9.8), below, is used to remove objects in the  $CM_4(x, y)$ .

$$CM_5(x, y) = \begin{cases} CM_4(x, y) & \text{if } Y_p > \eta \\ 0 & \text{otherwise} \end{cases} \quad (9.8)$$

where  $Y_p$  is the y axis position of each object in spatial space and  $\eta$  is a value set constant at 370 pixels. Searching of significant changes was limited from the ground to 1.5 m height, hence, any changes of trees and sky were not considered.

**9.2.4.2 Second Fuzzy Inference Sub-System**

Another Mamdani's fuzzy inference system, referred to as the second fuzzy inference system (FIS2), is developed in the second algorithm since object's areas and contours produced by the second algorithm are relatively different with object's areas and shapes provided by the first algorithm. A single number of the defuzzification output represents a change possible percentage of an object ( $P_b$ ). An overview of the FIS2 is depicted in Table 9.2, below.

Based on the  $P_b$  value, an object in the  $CM_5(x, y)$  is classified as an unimportant or a significant change by referring to (9.9) below.

$$CM_6(x, y) = \begin{cases} CM_5(x, y) & \text{if } P_b \geq \kappa \\ 0 & \text{otherwise} \end{cases} \quad (9.9)$$

where  $\kappa$  was set constant at 50 and the  $CM_6(x, y)$  is the second changed mask (CHM2\_II).



**Table 9.2** An overview of the FIS2 including its inputs, outputs, and fuzzy rules.  $I_1$ ,  $I_2$ ,  $I_3$  and  $I_4$  denote X-BoundingBox, Y-BoundingBox, Eccentricity and Xwidth.  $O$  symbolizes Probabilityaschange

No	Rules
1	$(I_3 == \text{high}) \Rightarrow (O = \text{low})$
2	$(I_1 == \text{low}) \& (I_2 == \text{low}) \& (I_3 == \text{high})$ $\Rightarrow (O = \text{low})$
3	$(I_1 == \text{low}) \& (I_2 == \text{high}) \& (I_3 == \text{low})$ $\& (I_4 == \text{small}) \Rightarrow (O = \text{low})$
4	$(I_1 == \text{high}) \& (I_2 == \text{low}) \& (I_3 == \text{low})$ $\& (I_4 == \text{small}) \Rightarrow (O = \text{low})$
5	$(I_1 == \text{high}) \& (I_2 == \text{high}) \& (I_3 == \text{low})$ $\& (I_4 == \text{small}) \Rightarrow (O = \text{high})$
6	$(I_1 == \text{high}) \& (I_2 == \text{high}) \& (I_3 == \text{low})$ $\& (I_4 == \text{large}) \Rightarrow (O = \text{high})$
7	$(I_1 == \text{high}) \& (I_2 == \text{low}) \& (I_3 == \text{low})$ $\& (I_4 == \text{large}) \Rightarrow (O = \text{high})$
8	$(I_1 == \text{low}) \& (I_2 == \text{high}) \& (I_3 == \text{low})$ $\& (I_4 == \text{large}) \Rightarrow (O = \text{high})$
9	$(I_1 == \text{very high}) \& (I_3 == \text{high})$ $\Rightarrow (O = \text{low})$
10	$(I_1 == \text{very high}) \& (I_3 == \text{low})$ $\Rightarrow (O = \text{low})$
11	$(I_1 == \text{low}) \Rightarrow (O = \text{low})$

As seen in Table 9.2, the FIS2 has four inputs (X-BoundingBox, Y-BoundingBox, Eccentricity and Xwidth), one output (Probabilityaschange) and 11 fuzzy rules. X-BoundingBox is interpreted as {Low, High, Very High} and Y-BoundingBox and

Eccentricity are interpreted as {Low, High}. Xwidth is denoted as {Small, Large}. The output is decoded as {Low, High}. Triangular membership functions (trimfs) were used for all inputs and the output. Parameters of the triangular membership functions are summarized in Table 9.3, below.

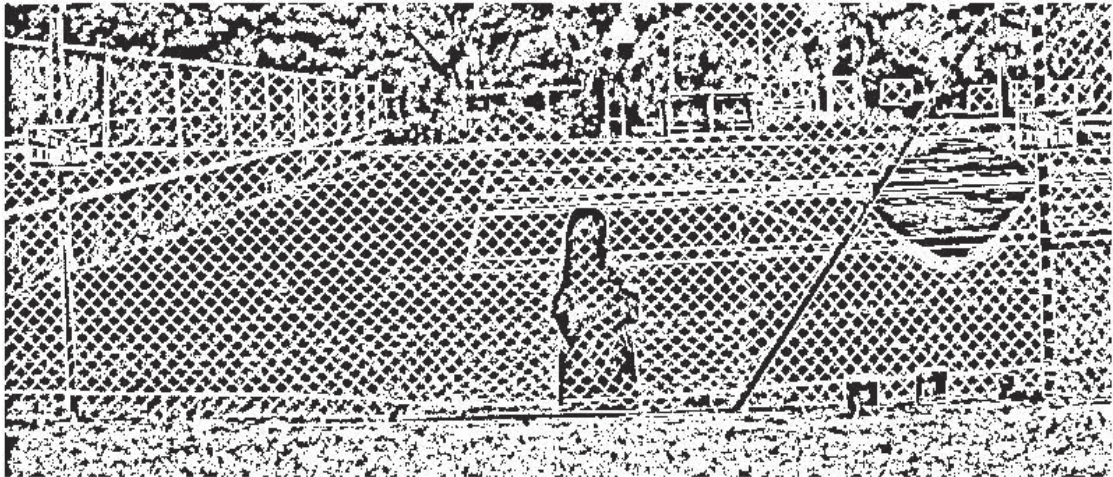
**Table 9.3** Summarization of triangular membership function parameters

	<b>trimfs</b>	<b>Parameters</b>
X-BoundingBox	Low	[-0.1 0 10]
	High	[10 1341 2985]
	Very High	[2985 3050 3089]
Y-BoundingBox	Low	[-0.8587 0 0.8587]
	High	[0.8587 572 1238]
Eccentricity	Low	[0 0.5 0.98]
	High	[0.98 0.99 1]
Xwidth	Small	[-312.4 0 312.4]
	Large	[312.4 1545 3089]
Probabilityaschange	Low	[0 25 50]
	High	[50 75 100]

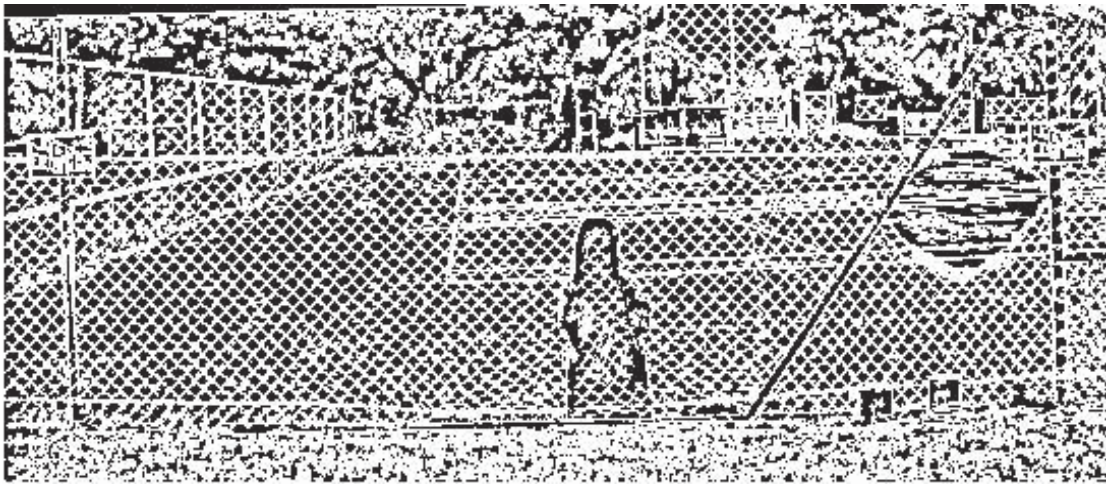
### 9.3 Experimental Results

The second algorithm uses each ROI2\_II as input. Prior to detect breaches in the integrity of and attached objects in front of fence wires, the algorithm firstly detects edges of fence wires in these kinds of multiple outdoor images. Results of detecting edges of fence wires are presented in binary images (Chapter 8), referred to as  $BW\_IIs(x, y)$ .

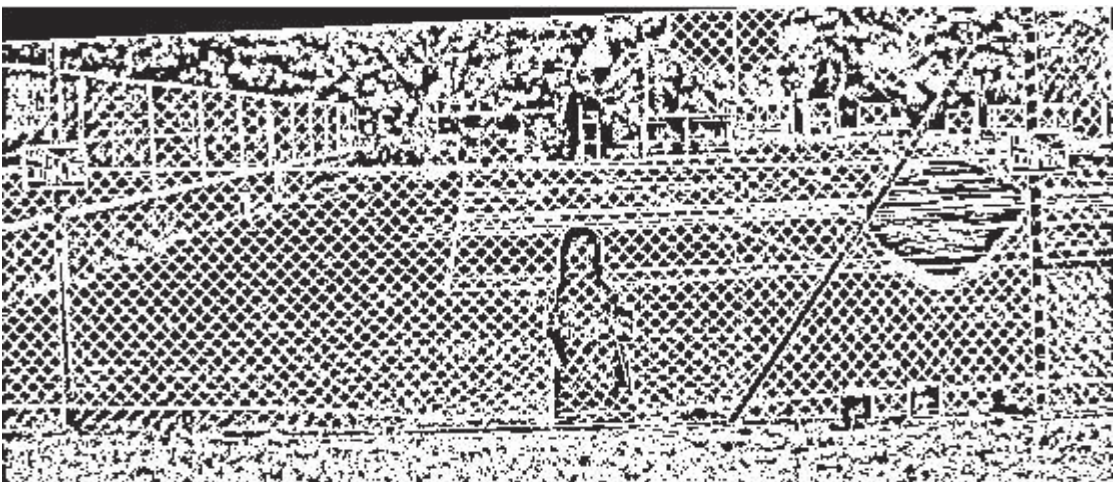
Edges of fence wires in every  $BW\_II$  are then enhanced in the next stage of the second algorithm. Morphological operations such dilation and erosion are used in the enhancing process. As a result, new binary images referred to as enhanced images ( $BW_3\_IIs$ ), are produced. The following Figs. depict enhanced images.



(a)

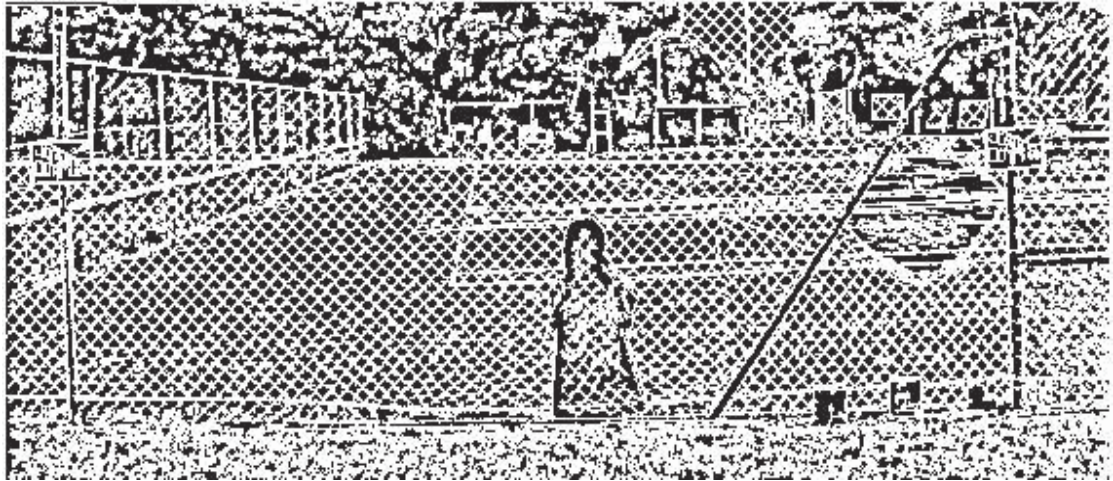


(b)

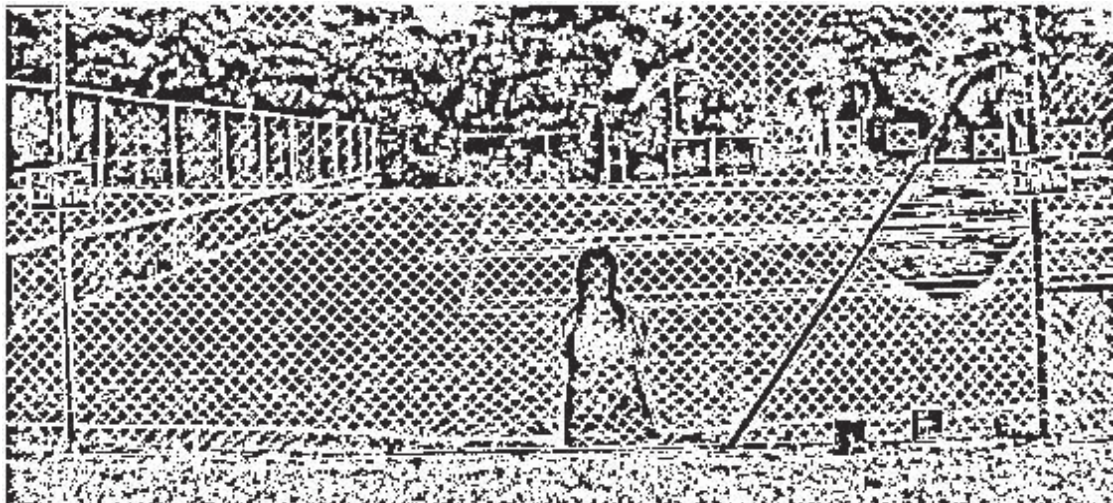


(c)

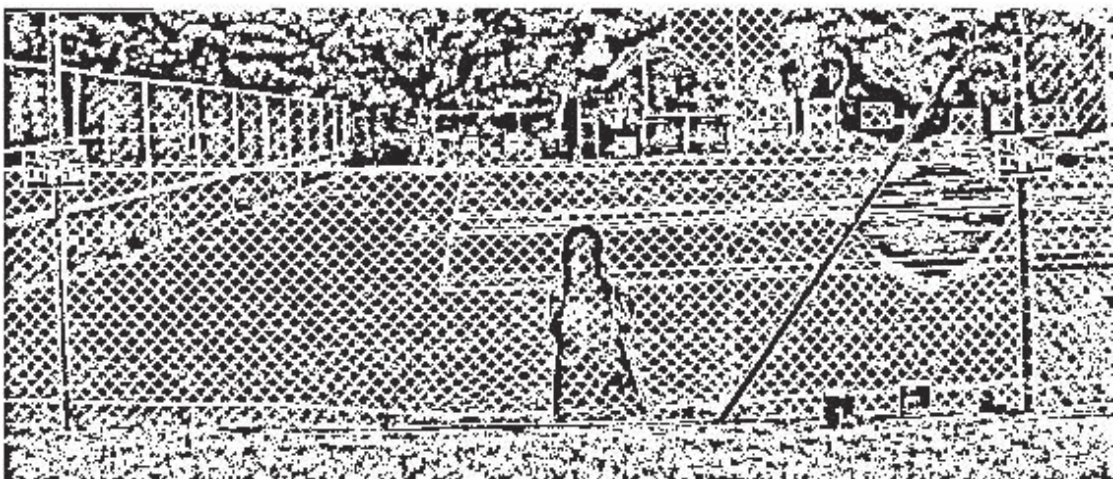
**Fig. 9.1** The  $BW_3_{II-1}$  (a),  $BW_3_{II-5}$  (b) and  $BW_3_{II-9}$  (c)



(a)

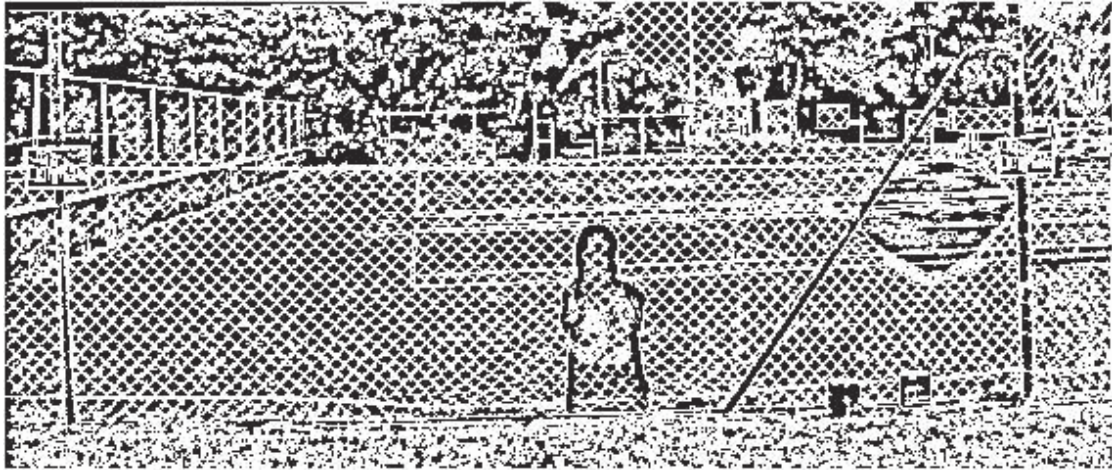


(b)

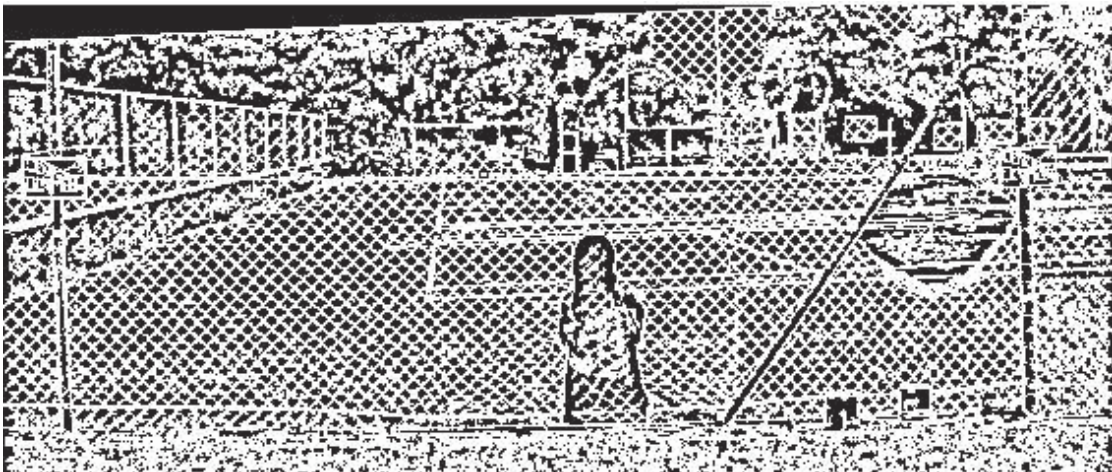


(c)

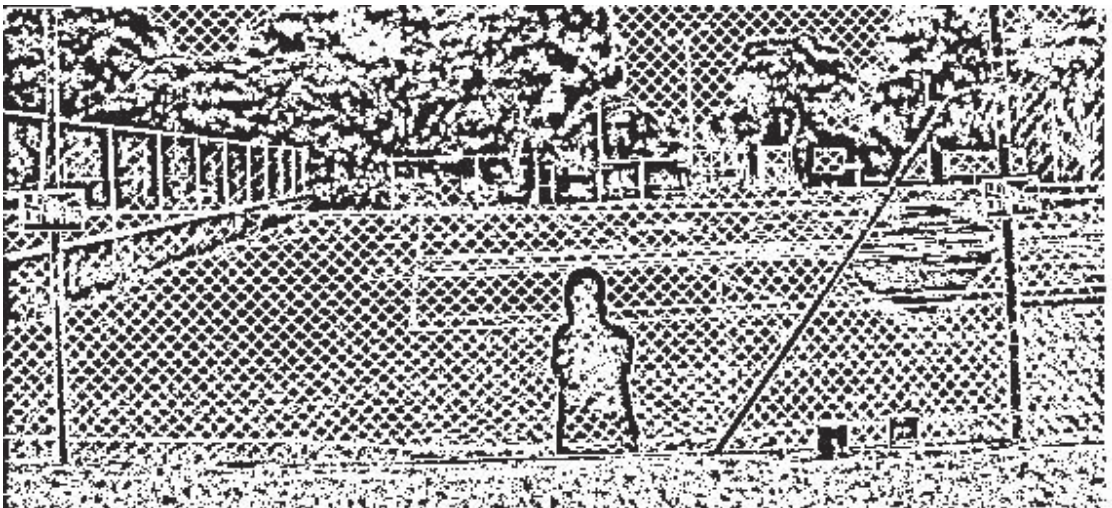
**Fig. 9.2** The  $BW_3_{II-2}$  (a),  $BW_3_{II-6}$  (b) and  $BW_3_{II-10}$  (c)



(a)

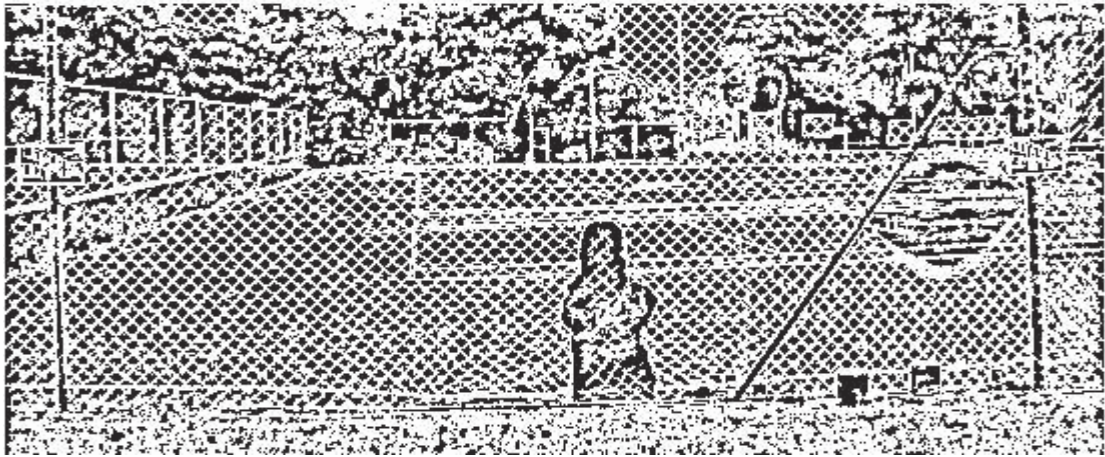


(b)

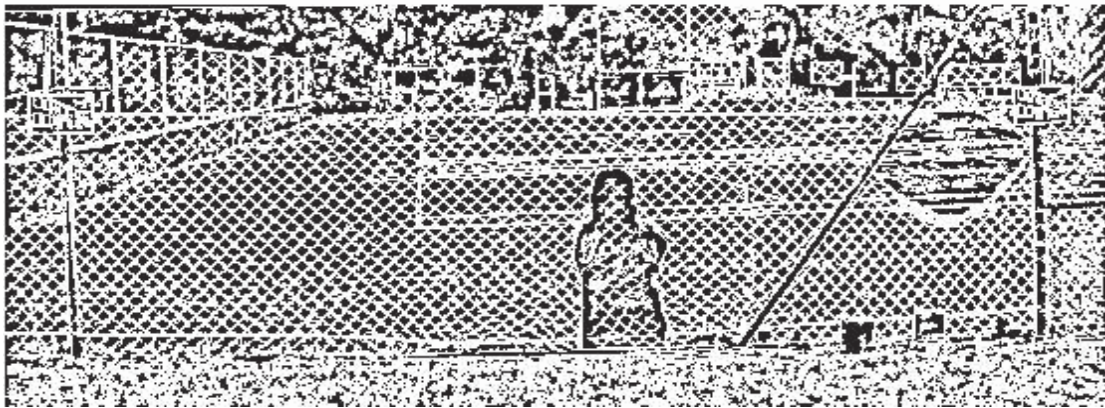


(c)

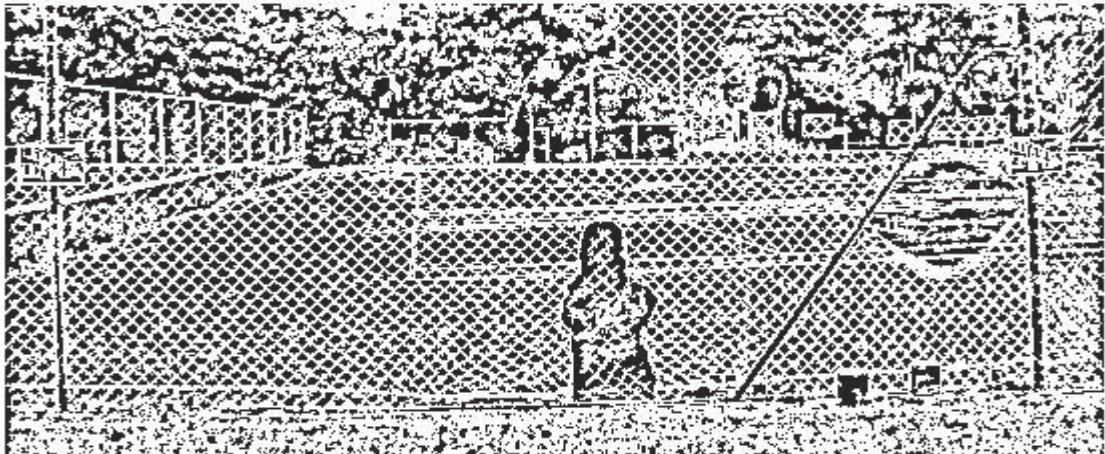
**Fig. 9.3** The  $BW_3_{II-3}$  (a),  $BW_3_{II-7}$  (b) and  $BW_3_{II-11}$  (c)



(a)



(b)



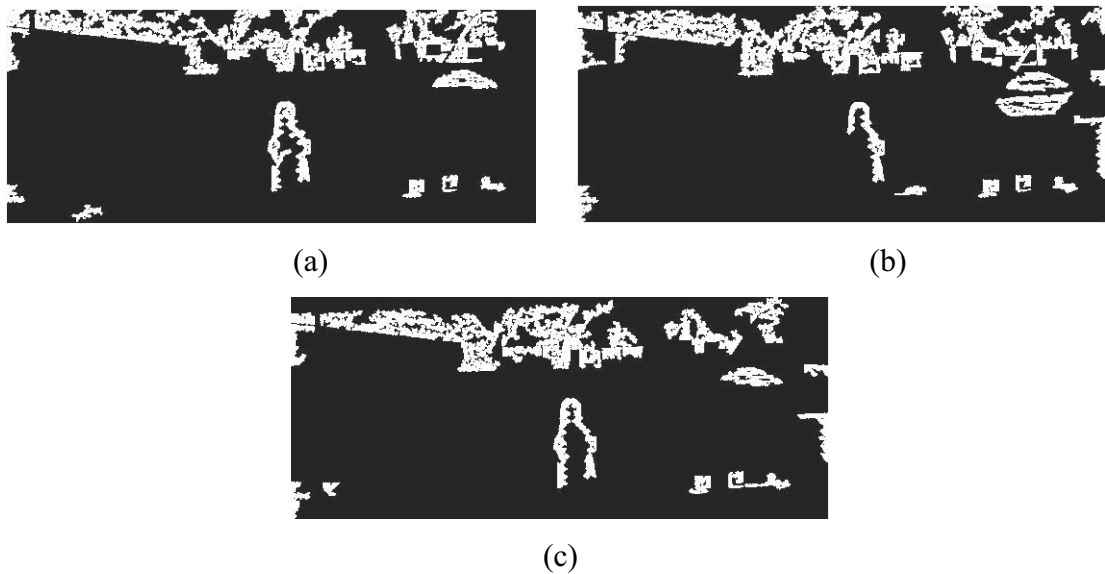
(c)

**Fig. 9.4** The  $BW_3_{II-4}$  (a),  $BW_3_{II-8}$  (b) and  $BW_3_{II-12}$  (c)

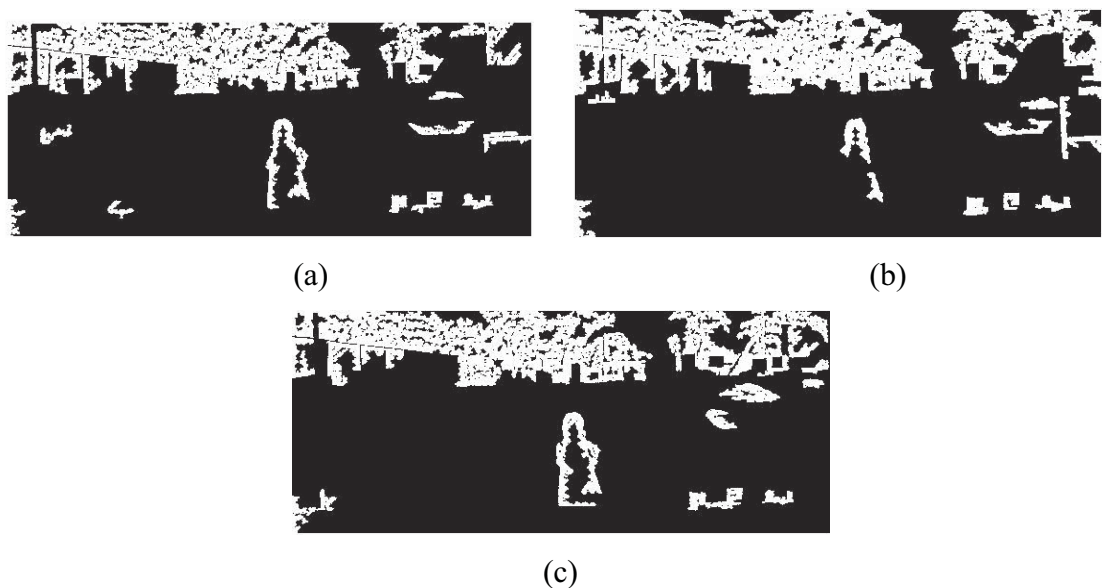
As seen in Figs. 9.1, 9.2, 9.3 and 9.4, above, edges of fence wires apparently appear in enhanced images. Edges of fence wires become thicker after performing the

enhancing process. As a result, diamond-shape darker areas observably emerge as well in enhanced images. By knowing the average area of these diamond-shape darker areas, potential significant changes can be extracted from enhanced images.

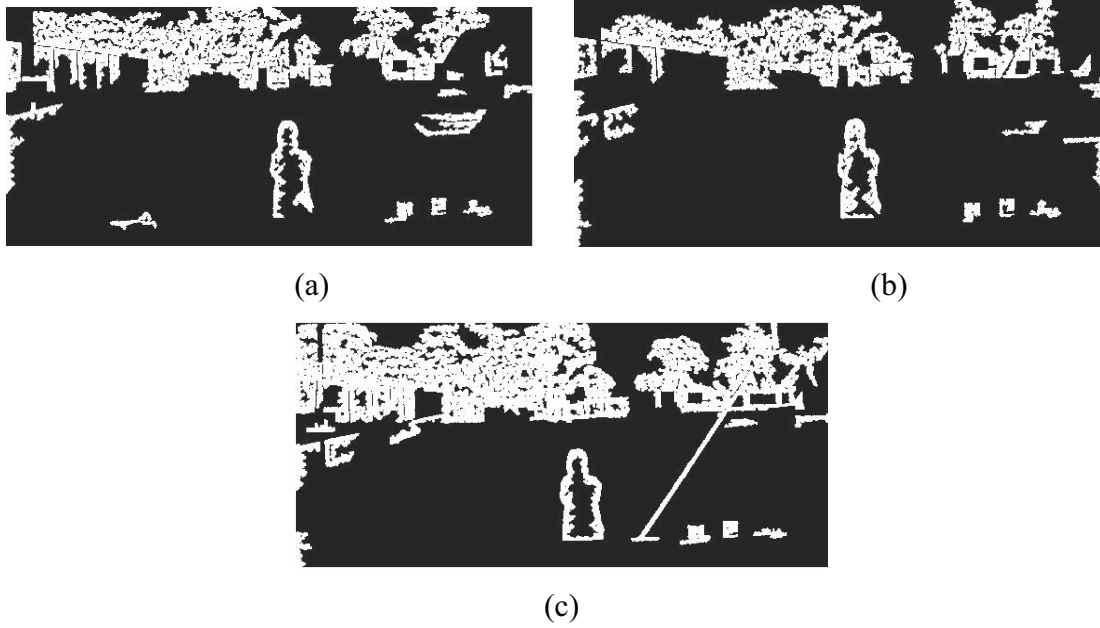
These diamond-shape darker areas are removed in the third stage of the second algorithm. The following Figs. depict results of applying steps in the third stage to enhanced images. New binary images, referred to as  $CM_4\_IIs$ , are produced by this third stage.



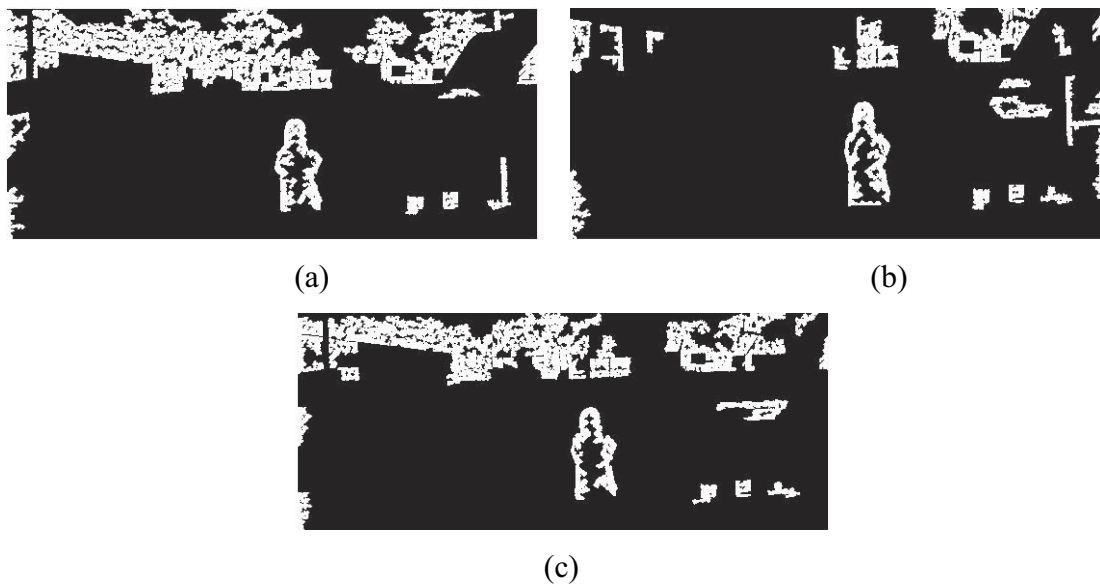
**Fig. 9.5** The  $CM_4\_II-1$  (a),  $CM_4\_II-5$  (b) and  $CM_4\_II-9$  (c)



**Fig. 9.6** The  $CM_4\_II-2$  (a),  $CM_4\_II-6$  (b) and  $CM_4\_II-10$  (c)



**Fig. 9.7** The  $CM_4_{II-3}$  (a),  $CM_4_{II-7}$  (b) and  $CM_4_{II-11}$  (c)

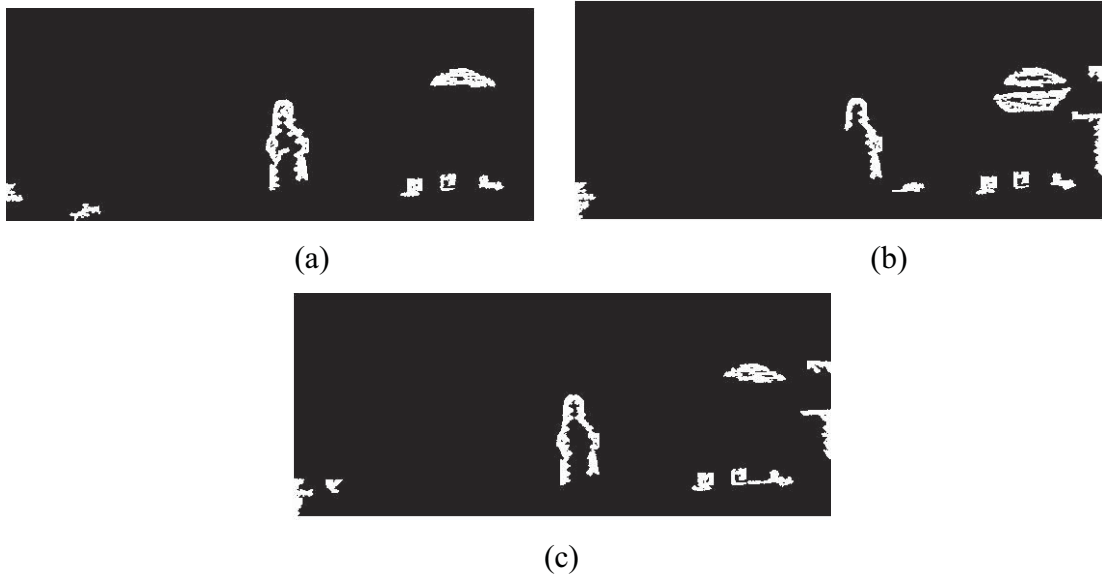


**Fig. 9.8** The  $CM_4_{II-4}$  (a),  $CM_4_{II-8}$  (b) and  $CM_4_{II-12}$  (c)

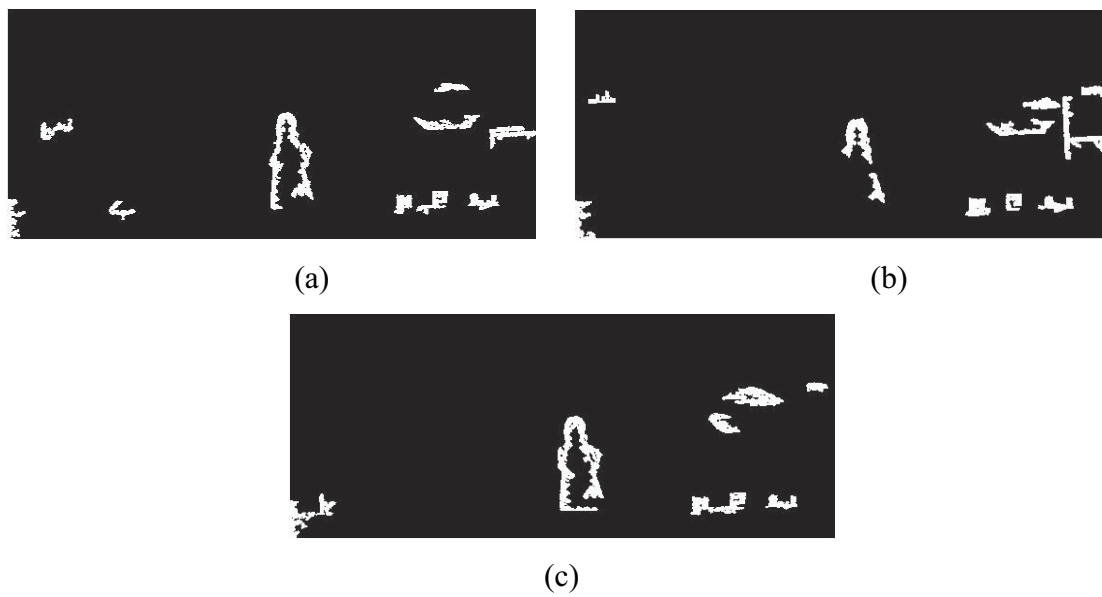
As depicted in Figs. 9.5, 9.6, 9.7 and 9.8 above, diamond-shape areas have been removed and objects from the ground to 1.5 meters height on the background have been eliminated as well. However, objects above 1.5 meters height on the background such as trees and buildings still appear in each  $CM_4_{II}$ .

To remove such trees and buildings, a crisp inference system is applied to every  $CM_4_{II}$ . The following Figs. depict results of applying the crisp inference system.



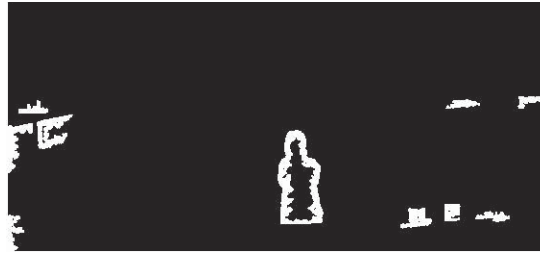


**Fig. 9.9** The  $CM_5_{II-1}$  (a),  $CM_5_{II-5}$  (b) and  $CM_5_{II-9}$  (c)



**Fig. 9.10** The  $CM_5_{II-2}$  (a),  $CM_5_{II-6}$  (b) and  $CM_5_{II-10}$  (c)



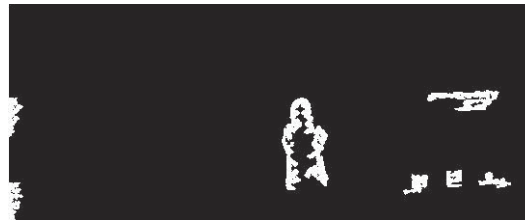


(c)

**Fig. 9.11** The  $CM_5\_II-3$  (a),  $CM_5\_II-7$  (b) and  $CM_5\_II-11$  (c)

(a)

(b)



(c)

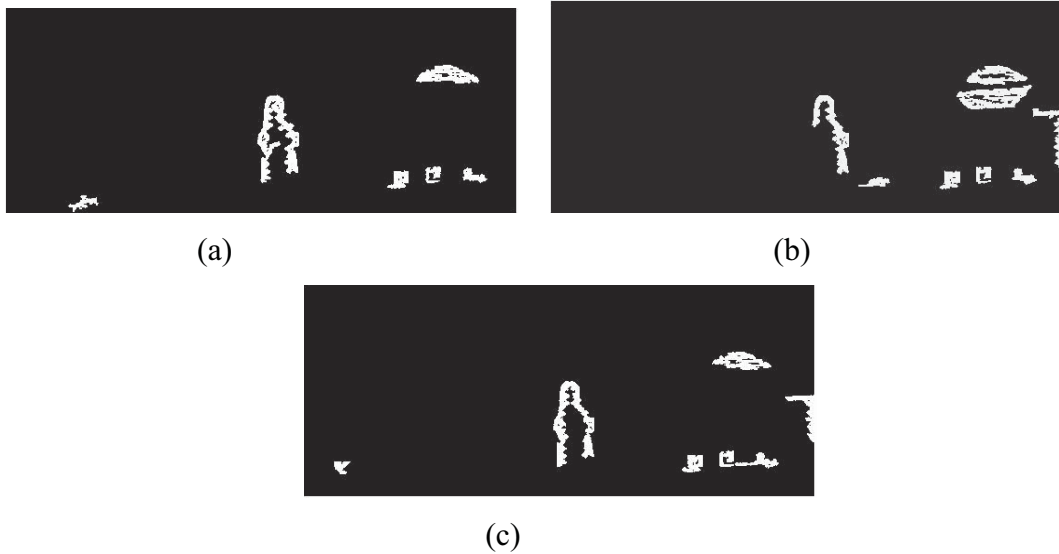
**Fig. 9.12** The  $CM_5\_II-4$  (a),  $CM_5\_II-8$  (b) and  $CM_5\_II-12$  (c)

As seen in Figs. 9.9, 9.10, 9.11 and 9.12 above, objects on the background such trees and buildings have been removed by the crisp inference system. However, there are still a few unimportant changes observably appear in each  $CM_5\_II$ .

Finally, the FIS2 is applied to each  $CM_5\_II$  in order to generate a final changed mask (CHM2\_II). All CHM2\_IIs is presented in the next section.

#### 9.4 Changed Masks

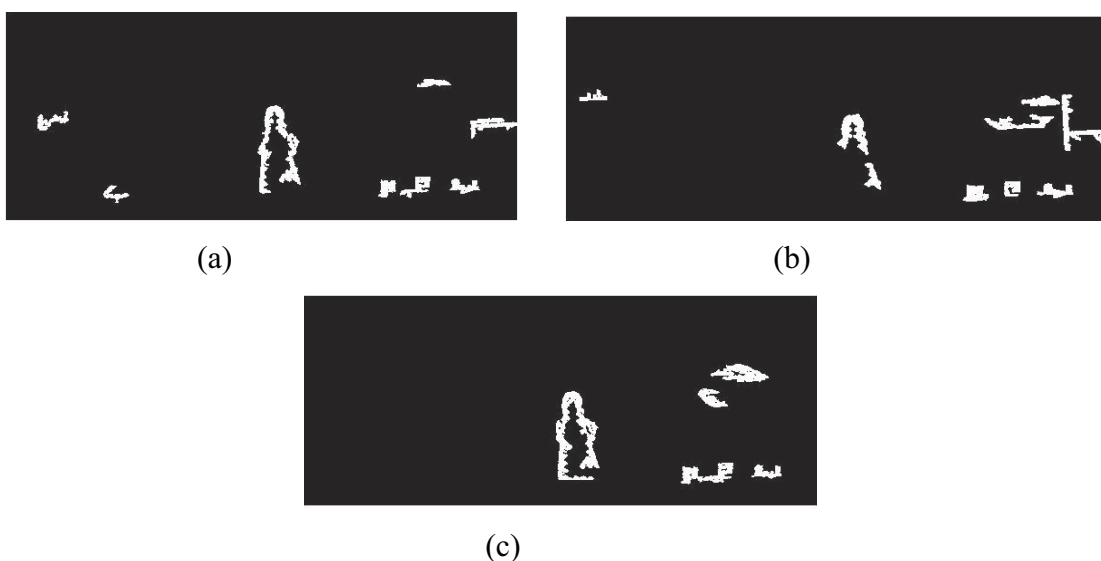
Changed masks detected by the second algorithm from input images are presented in this section. Figs. 9.13 (a), 9.13 (b) and 9.13 (c) depict changed masks produced by the second algorithm from the ROI2\_II-1, ROI2\_II-5 and ROI2\_II-9.



**Fig. 9.13** Changed masks extracted by the second algorithm from the ROI2\_II-1 (a), ROI2\_II-5 (b) and ROI2\_II-9 (c)

As seen in Figs. 9.13 (a), 9.13 (b) and 9.13 (c), all significant changes (i.e., a large breach in the right top of fence wires, a small breach in the right bottom of fence wires and two attached objects in front of fence wires) are correctly detected by the second algorithm.. However, there are a few unimportant changes that observably appear in the changed masks, above.

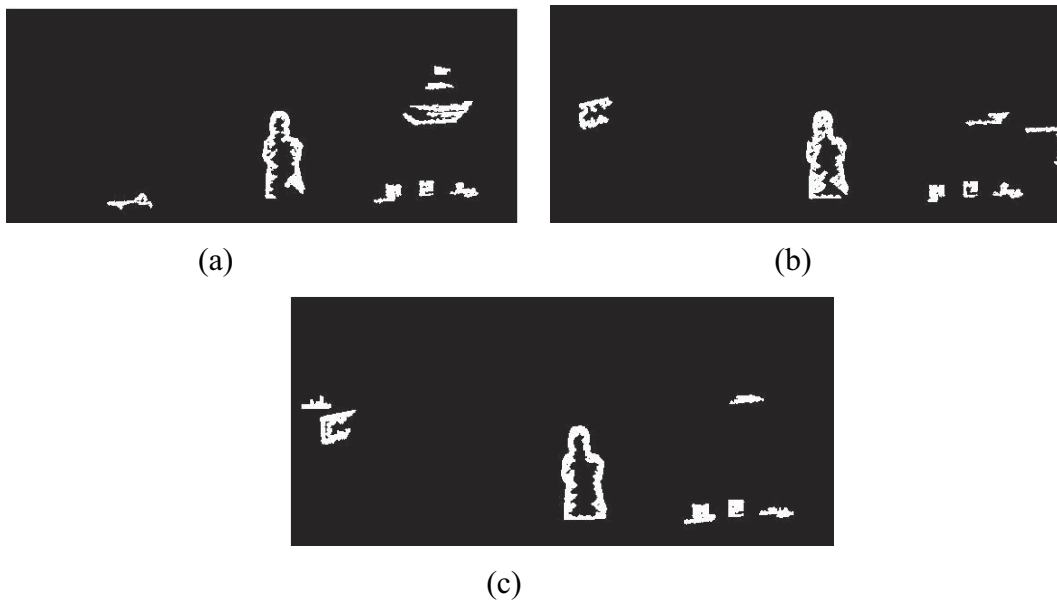
Figs. 9.14 (a), 9.14 (b) and 9.14 (c) depict changed masks produced by the second algorithm from the ROI2\_II-2, ROI2\_II-6 and ROI2\_II-10.



**Fig. 9.14** Changed masks detected by the second algorithm from the ROI2\_II-2 (a), ROI2\_II-6 (b) and ROI2\_II-10 (c)

As can be seen in Figs. 9.14 (a), 9.14 (b) and 9.14 (c), all significant changes has been successfully detected by the second algorithm from the ROI2\_II-2, ROI2\_II-6 and ROI2\_II-10. However, several unimportant changes observably appear as well. The unimportant changes are trees on background of the outdoor scene, the grass and right post of fence wires.

Changed masks generated by the second algorithm from the ROI2\_II-3, ROI2\_II-7 and ROI2\_II-11 are depicted in Figs. 9.15 (a), 9.15 (b) and 9.15 (c).



**Fig. 9.15** Changed masks produced by the second algorithm from the ROI2\_II-3 (a), ROI2\_II-7 (b) and ROI2\_II-11 (c)

As seen in Figs. 9.15 (a), 9.15 (b) and 9.15 (c), the second algorithm can detect all potential significant changes. It also detects the intruder which is close to its original contour. Moreover, several unimportant changes observably come out as well.

Figs. 9.16 (a), 9.16 (b) and 9.16 (c) depict changed masks detected by the second algorithm from the ROI2\_II-4, ROI2\_II-8 and ROI2\_II-12.





(c)

**Fig. 9.16** Changed masks extracted by the second algorithm from the ROI2\_II-4 (a), ROI2\_II-8 (b) and ROI2\_II (c)

As can be seen in Figs. 9.16 (a), 9.16 (b) and 9.16 (c), the second algorithm has been successful in detecting all locations of potential significant changes. An unimportant change is also detected by the second algorithm (see Fig. 9.16 (b)). The unimportant change is connected edges of right post of fence wires and the ROI2\_II-8's image.

### 9.5 Concluding Remarks

The second algorithm is designed to detect only breaches in the integrity of and attached objects in front of fence wires. The minimum size of a breach that can be detected by the second algorithm is the dimension of 15 cm by 10 cm when an 8MB image captured from 600 cm distance from the digital camera to the scene. The minimum size of an attached object detected by the second algorithm is the dimension of 11 cm by 11 cm.

The second algorithm is composed by four main stages. (1) Detecting edges of fence wires using the Sobel edge detector combined with an adaptive thresholding technique. The combined Sobel detector and adaptive thresholding technique has a major role in the second algorithm since it is used to detect edges of fence wires and edges of attached objects in front of fence wires without confusion from specular reflections that occur on their surfaces. (2) Enhancing edges of fence wires using morphological operations such as dilation and erosion. (3) Removing diamond-shape areas using a simple area-based algorithm. Since a diamond netting fence is used in this research, diamond-shape objects will appear in between fence wires. Areas of breaches and attached objects will be different with areas of diamond-shape objects. By knowing an average area of diamond-shape objects, breaches and attached objects can be detected. (4) Determining which objects in each changed masks as significant

or unimportant changes using the second hybrid decision-making system. It consists of crisp and fuzzy inference systems.

Changed masks produced by the second algorithm show that it detects correctly all significant changes from input images. Although it detects significant changes in input images, it also detects the intruder and several unimportant changes. Since these kinds of multiple outdoor images are very complex as a consequence of background clutter, illumination variation, camera motion, thinness of fence wires and non-uniform illumination that occurs across fence wires, the second algorithm has been successful to minimize unimportant changes that arise in these kinds of multiple outdoor images.

Since there are many kinds of wire fences in the market such as taut wire fences, chain-link mesh fences (diamond netting fences), welded mesh fences and galvanized razor wire fences, the second algorithm is only suitable applied to diamond netting fences and welded mesh fences. The third main stage of the second algorithm, removing diamond-shape areas using a simple area-based algorithm, is the main reason of this limitation. The area-based algorithm works by finding firstly an average size of diamond-shape objects in between fence wires performed once in the calibration process. Since sizes of breaches in the integrity of and attached objects in front of fence wires are quite different from sizes of diamond-shape objects, diamond-shape objects are finally removed based on the average size.

Since welded mesh fences have a similar structure with chain-link mesh fences, the second algorithm is applicable for welded mesh fences. In the case of welded mesh fences, square- or rectangle-shape objects will apparently appear in between fence wires instead of diamond-shape objects. By knowing an average size of square- or rectangle- shape objects acquired in the calibration process, breaches and attached objects will be able to be identified.