Washington University in St. Louis

School of Engineering and Applied Science

Electrical and Systems Engineering Department

ESE498

Intruder Detection System

By

Allen Chiang, Jonathan Chu, Siwei Su

Supervisor

Professor Robert Morley

Submitted in Partial Fulfillment of the Requirement for the BSEE

Degree,

Electrical and Systems Engineering Department, School of

Engineering and Applied Science,

Washington University in St. Louis

May 2014

Student Statement

This project is the official documentation for the Intruder Detection System built by Senior Electrical Engineering students Allen Chiang, Jonathan Chu, and Siwei Su. The three students are the main designers of the project, and have maintained the integrity of the course in the design of this project. These designs are in compliance of the Washington University Honor Code.

Project Abstract

In this project, we used LabVIEW with NI Vision to build an intruder detection system algorithm to detect the presence of an intruder. We realized that household pets could be a problem in most household safety systems, as their movement could easily trigger it off. It is because of this that we designed our algorithm to ignore any pets that come into the viewing space. Our system alerts the user through SMS and Email if any intruder has been detected. The images that were acquired and analyzed through frame comparisons converted to binary images that represent change, and then filtered through a series of image refining VI's, helping to enhance our change detection results.

Acknowledgements

Table of Contents

List of Figures

List of Tables

Problem Formulation

Problem Statement:

According to the FBI, 97% of all home intrusions happen when no one is at home. In 2012, a home was invaded every 20 seconds (burglarized or robbed). Around the United States, 4457 homes are robbed or burglarized every single day. However, only 18% of these incidents end in arrests.

There is an increasing problem of home intruders in the United States. More and more people are finding ways to break into homes, posing a safety threat to homeowners and neighborhoods. People turn to crime, thinking that the risk is worth the return, since many intruders never get caught. This creates a very bad environment for people to live in, as they would constantly have to live in fear. The problem with home security in the modern world is a cause for concern. In the United States in 1998, police responded to approximately 38 million home security alarm activations, at an annual cost of approximately $1.5 billion. Nationwide, false alarms would by itself account for 10 to 25 percent of all calls to the Police Department. That is between 3,800,000 and 9,500,000, costing between $150 million and $375 million. Among these false alarms, many are triggered due to owners accidentally setting the alarm while household pets are free to roam.

We wanted to fix this problem, by building a home intruder detection system that can accurately detect a human intruder, while filtering out movements that come from household pets.

The self-proposed problem to be solved in our project is to design and implement an intruder detection system.

The objectives are:

- the system should be able to alert the user once an intruder has been detected
- the system should be able to differentiate between a human intruder and a household pet
- the system should apply filters to minimize false positives and false negatives
- the system should work in any enclosed environment (such as a living room, dorm room, etc.)
- the system should be reliable

The constraints are:

- Costs. We will try and create our system with minimal costs.
- We will create the system using software that is accessible to us for free through the engineering school

Problem Formulation:

- We decided to use LabView to tackle our problem of creating an intruder detection system. We chose LabView because it is accessible for free through the engineering school, it contains a NI Vision package that contains many useful VIs that we can utilize, it is known for being a useful software for prototyping, and because it is compatible with a wide variety of webcams.
- The verification of the accuracy and success of our system will be very straightforward at the end of the project: the system will either be able to detect human intruders or it will be unable to.

Project Specifications

Our design will include the following:

Image acquisition:

- Our camera is able to capture images of various resolutions and image types. The resolution and image type that we choose to acquire is a 800 x 600 RGB24 video input.

- Our camera is able to capture frames at a maximum rate of 35 frames per second for 800 x 600 RGB24 video input.

Image processing:

- Must compare consecutive frames by subtracting pixels to find the area of change. The system must subsequently convert the subtracted image to a binary image.

- Must actively filter binary images using a "salt and pepper filter" to filter out noise.

- Must improve binary image quality by rounding edges and filling in empty spaces.

Intruder Algorithm:

- Extract the pixel height of the area of change determined by our image processing workflow, compare the pixel height with a threshold. The threshold is a predetermined value that is dependent on the area of enclosed space that the camera is capturing.

- Extract the width and height of the area of change determined by our image processing workflow. Compare the ratio of the width and height of the area of change to a predetermined ratio threshold.

Email SMS Alert:

- Actively sends out alerts every 3 seconds while intruder is inside the camera view

- Must halt SMS/Email operations when intruder steps out of camera view

- System must do nothing in the event of a pet

System operation:

- Accuracy in detections(of intruders or pets) of up to 90%

- Using 800 x 600 RGB24 video input, our system runs with a frame rate of around 25 frames per second.

Concept Synthesis

Literature Review:

- An IEEE paper, "Image change detection algorithms: a systematic survey", by Radke, R.J., Andra, S., Al-Kofahi, O., and Roysam, B., gave us many ideas and potential algorithms for detecting change while we were brainstorming for methods for detecting change in images. The paper showed us various ways to improve change detection, such as preprocessing models, predictive models, and significance and hypothesis testing.

- A website with an introduction and tutorial to image change detection, "Tracy's Creative Image Tutorials", had very clear and easy to understand step by step images that brings you through a sequence of images showing what was done and how it contributes to finding the change in an image. The website was very helpful for understanding the gist for image change detection in the early stages of our design process.

- Our group started the project with no prior experience with LabView, thus a lot of our reading was directed to the learning of LabView. The IMAQ Vision for LabVIEW User Manual and other LabVIEW Manuals helped us get started with coding in LabVIEW.

Concept Generation

- In the design of a reliable intruder detection system, we had 2 main components of the project: Detection Method and Functionalities. We had to ensure that we could build a system that could reliably detect intruders as well as perform functions to alert and process the images. In the beginning, we were considering various detection techniques, such as

weight detection or even a simple password key input system. Below is a tree diagram illustrating our brainstorming process and ideas.

Figure 1. Tree diagram illustrating our brainstorming process and ideas

Concept reduction

In the choice between our different methods for detecting an intruder, we ruled out detecting an intruder with a proximity sensor due to the simplicity of a proximity sensor. A proximity sensor would not offer us as much customizability as either weight sensors or using a webcam for image processing. For example the proximity sensor can only sense when something is detected, however it can't provide you with details regarding what was detected, thus failing the requirement of minimizing false positives and false negatives.

We decided to rule out detecting an intruder through weight detection when we realized that this method is not reliable enough. The idea of weight detection is to place a pressure sensor at the entry points of the enclosed space, such as doors or windows. The problem with weight sensors, however, is that an intruder can easily step over and avoid the sensor, therefore rendering the intruder detection system useless.

In the end, we decided that using a camera and creating an intruder detection system with image processing would be best for our project, as it offers customizability and is much harder for an intruder to avoid. The customizability of image processing will let us apply filters and functions to improve the accuracy of the system, thus satisfying the requirement of minimizing false positives and false negatives. Additionally, through image processing we will also be able to differentiate between pets or humans. Upon making the decision to deal with images, we had to order a reliable (and cost-efficient) camera that could process our images with a reasonable frame rate.

Detailed Engineering Analysis and Design Presentation

The camera captures a real time video stream. Then the video is fed into Labview for processing. The difference between the current frame of the video and its previous frame is analyzed. If there is an intruder in the video, an alert email and text message will be sent by Labview.

Figure 2. Structure and data flow of Intruder Detection System. Processing occurs in three steps: signal acquisition, signal processing, and alert.

Image Acquisition

The camera keeps acquiring the video feed in real time. The IMAQdx Open Camera VI opens the camera and the IMAQdx Configure Grab VI configures and starts a grab acquisition. The IMAQdx Grab VI acquires the most current frame.
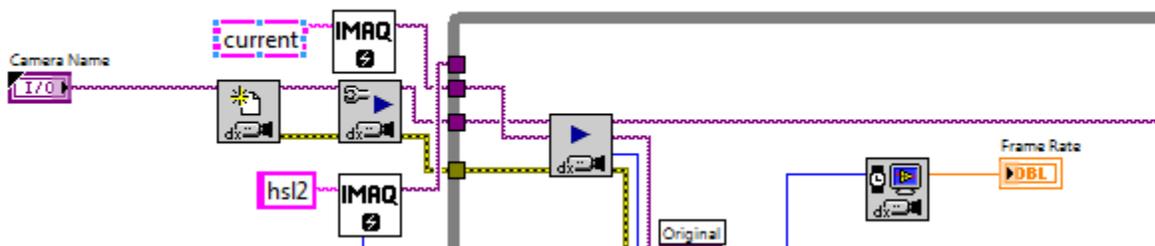

Figure 3. Labview VIs used to acquire images: IMAQdx Open Camera, IMAQdx Configure Grab, IMAQdx Grab

Figure 4. Original Image Acquired from Camera

Image Processing

1) Subtraction

The current frame of the video is subtracted by its previous frame using the IMAQ Subtract VI. If there is an intruder in the video, there will be a difference between two consecutive frames. If the intruder is constantly moving, the difference is the shape of the intruder.
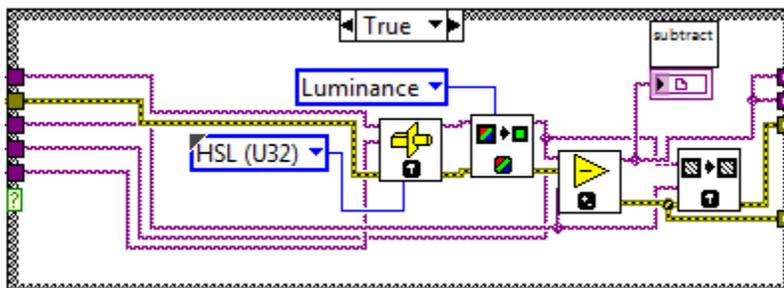

Figure 5. The part of Labview code doing subtraction

2) Threshold Image

Turn the subtracted image into a binary image by using the IMAQ Threshold VI so that each pixel will have a value of either 1 or 0. Then we can filter out unwanted noises easily.
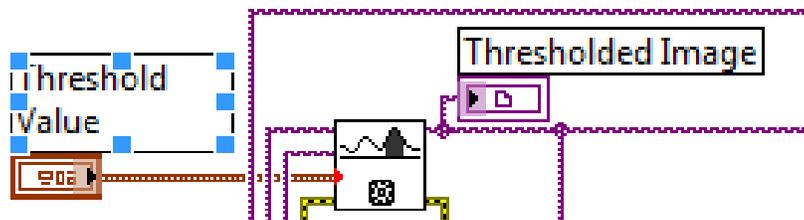
Figure 8. Image converted into binary image

3) Filter Out Noises and Fill Holes

Apply the IMAQ RemoveParticle VI and the IMAQ Morphology VI to the binary image to filter out noises, which are small particles. Then apply the IMAQ Convex Hull VI to fill the holes.
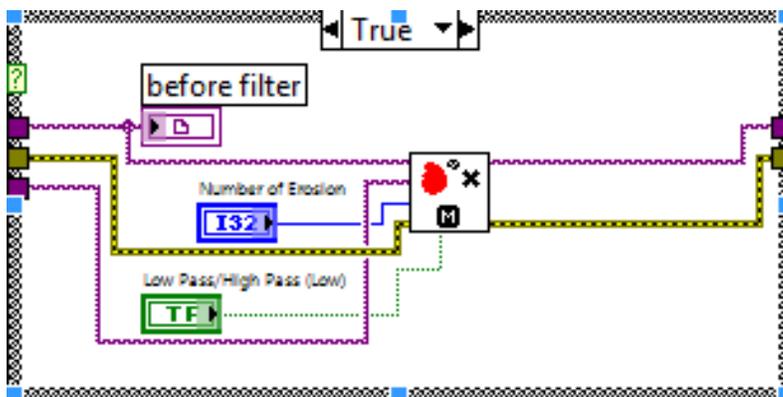

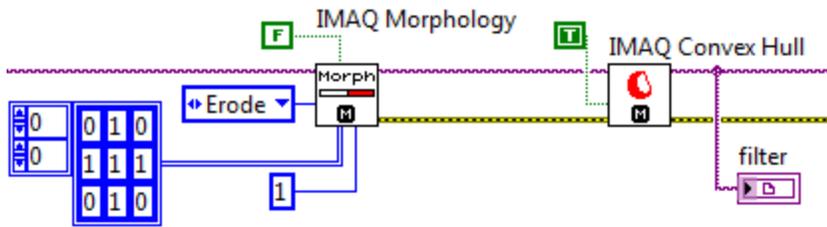Figure 9. The part of code doing filtering

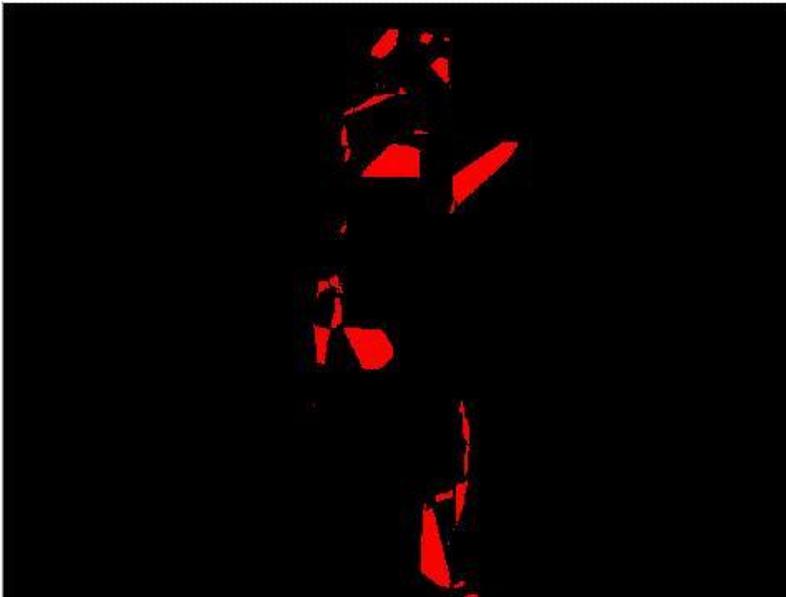Figure 10. The part of code doing Morphology and filling holes

Figure 11. Image after filtering

4) Clamp the Filtered Image

Apply horizontal clamp and vertical clamp to the filtered image to get the height and width of the figure in the image.
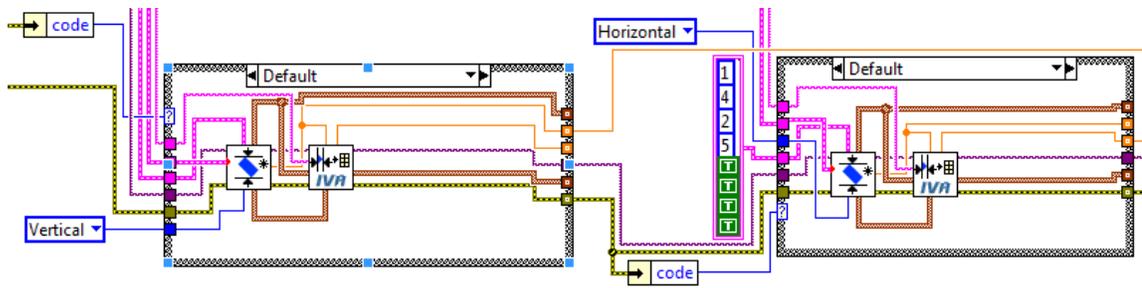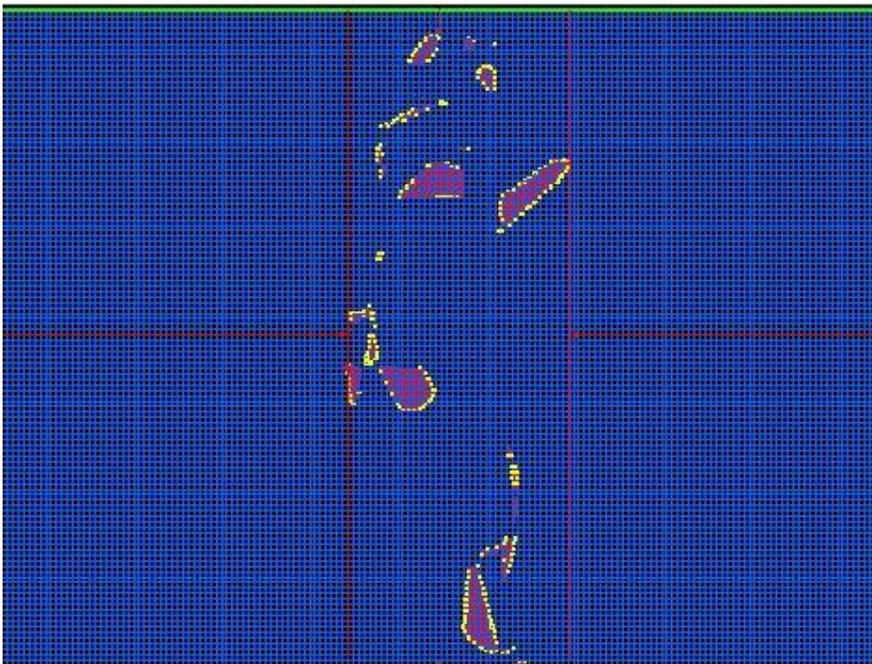
Figure 12. The part of code doing clamping

Figure 13. Image after clamped

5) Draw Tracking Box

Draw a box around the moving object based on its height and width. Then overlay the tracking box on the original image for visual appeal to users.
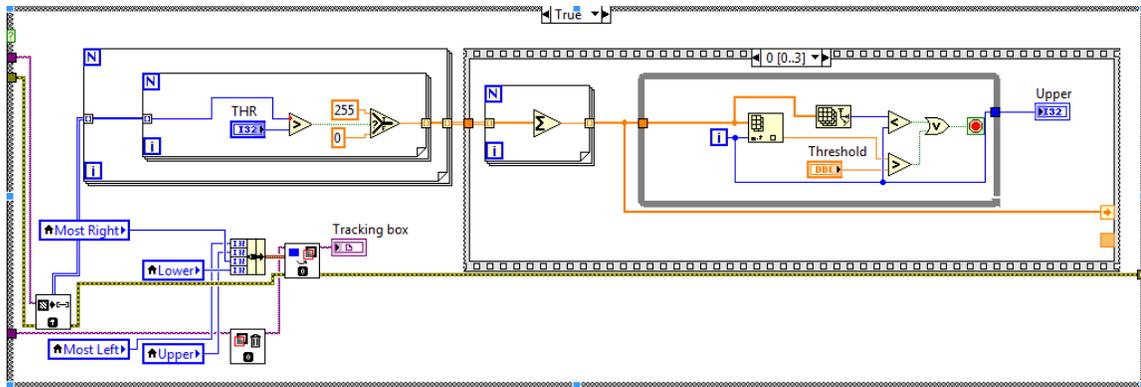
Figure 14. The part of code drawing tracking box

Figure 15. The original image with tracking box

Alert

The height given by the horizontal clamp is first compared with a range of threshold set by designers, which checks if the moving object is a pet. If the height is high enough to fall out of the range, the height is then divided by the width given by the vertical clamp to get the height-width ratio. The ratio is then compared with another range of threshold set by designers, which is the possible

range of human height-width ratio. If the calculated ratio falls into the range, the alarm will be triggered and alerting emails and text messages will be sent to the users.
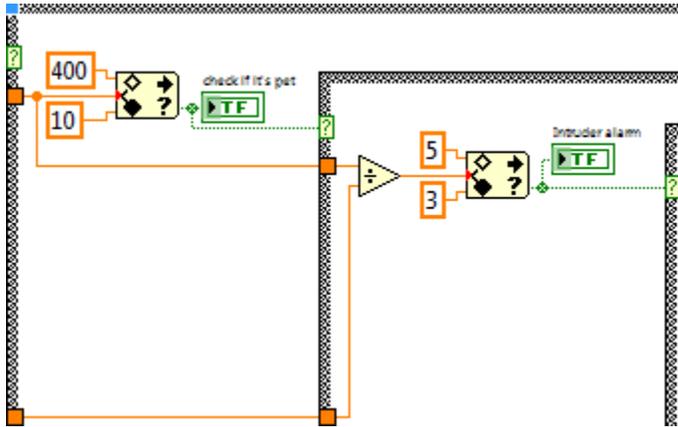

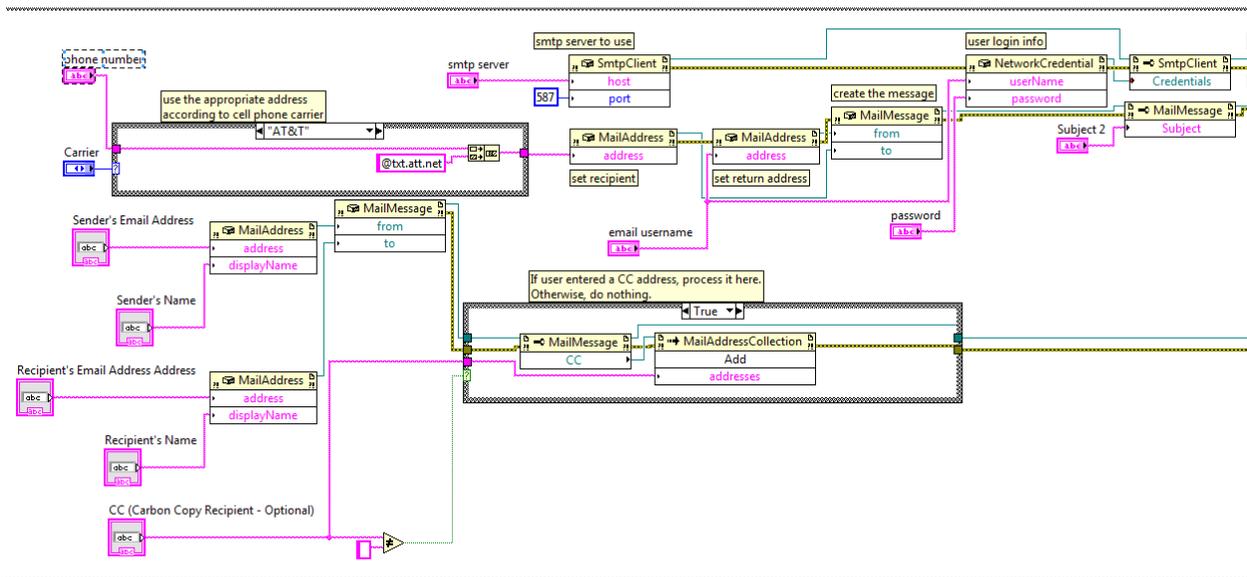**Figure 16. The thresholds used to check if the moving object is a pet or a human**

Figure 17. The part of code sending emails and text messages

Cost Analysis

For a user already with LabView installed with NI Vision, the costs of implementing our Intruder Detection System is just the cost of the webcam, which can be any economic webcam and can be easily bought for under $20.

For a user without LabView, the user would have to purchase LabView (as low as $20 for students and up to $1,249 for the full price). Considering the price of NI Vision, the total price of our system can easily go for several thousand dollars. As we can see, this system would not be affordable if the user did not already have LabView and NI Vision.

However for further investigation, if our code can be packaged into an executable software, then our intruder detection system can be priced at an extremely affordable price and could be easily sold online for download.

Component	Supplier	Item (Item No.)	Cost for Prototype	Cost for Production (per unit)	
Web Camera	Sabrent	BPC3-CAMWCM6LNV	$19.98	Undecided	http://www.bestpricecables.com/sabrent-usb-2.0-web-camera-6-led-infrared-night-vision-w-audio-mi.html
Total			$19.98		

Table 1. Bill of Materials

Hazards and Failure Analysis:

The user should follow all safety instructions that come with their webcam and computer. Our intruder detection system does not come with any safety or health hazards.

Conclusion

In the end, we were able to design a reliable intruder detection system that was able to differentiate between humans and pets. The system utilized set thresholds and filters to minimize rates of false positives and negatives. The system is consistent, accurate, and reliable, with an accuracy of over 90%.

Further investigations include incorporating night vision functionality as well as sound actuators to dissuade intruders. Implementing multiple cameras may also increase viewing area, and thus, improving functionality. We would also need to find a method to package system to increase portability. Additionally, our current system must request user to manually set the threshold values. These threshold values vary, depending on the depth of the camera viewing space. For further investigation, an automated threshold algorithm could be developed to detect the depth of the image and automatically set a corresponding threshold as a result.

References

1. Radke, R.J.; Andra, S.; Al-Kofahi, O.; Roysam, B., "Image change detection algorithms: a systematic survey," Image Processing, IEEE Transactions on , vol.14, no.3, pp.294,307, March 2005

2. Radke, R.J.; Andra, S.; Al-Kofahi, O.; Roysam, B., "Image change detection algorithms: a systematic survey," Image Processing, IEEE Transactions on , vol.14, no.3, pp.294,307, March 2005

3. Rose, T , "Change Detection Images". RetrievedApril , 2014 Available: http://www.tracyleerose.com/Txt/Tutorials/Tutorial_Changedetection/Changedetection.html

4. Sampson, R , "False Burgler ". RetrievedApril , 2014 Available: http://www.cops.usdoj.gov/pdf/e05021556.pdf

5. "IMAQ Vision for LabVIEWTM User Manual". RetrievedApril , 2014 Available: http://www.ni.com/pdf/manuals/371007a.pdf

6. "Image Filtering Overview". RetrievedApril , 2014 Available: http://www.ni.com/white-paper/3472/en/

7. "FBI Statistics". RetrievedApril , 2014 Available: http://patriotcrimedefense.com/fbi-statistics

8. "Improve an Image". RetrievedApril , 2014 Available: http://zone.ni.com/reference/en-XX/help/370281M-01/nivisionlvbasics/improve_an_image/