

Research Article

An Improved and Secure Motion Detection Surveillance System in UNIX

Syed Zahurul^{1,*}, Razali Jidin¹, Norashidah Md. Din¹, Syed Zahidul¹

(1) Electrical & Communication Engineering, Universiti Tenaga Nasional, Malaysia.

Copyright © 2013 Australian International Academic Centre, Australia

doi:10.7575/aiac.ijaep.v.1n.2p.13

Article history:

Received 02 October 2013
Reviewed 08 October 2013
Revised 10 October 2013
Accepted 13 October 2013
Published 14 October 2013

Abstract. *These days motion detection in surveillance technology has a trend to deploy to security sectors using radar motion detector, wireless sensors, Micro-Electro-Mechanical Systems (MEMS) Inertial Sensors etc in Unix system. This research focuses on motion detection using combination of optical flow and thresholding methods on Linux platform. Salient configurations, inevitable installation packages and libraries are discussed elaborately. Moreover, script with pseudo code for connecting with remote networking service application has been implemented. Results show a good and convenient security system suitably for home and offices.*

Keywords: motion detection, unmanned aircraft, video surveillance, optical flow model, gradient-based optical flow, subtraction Model.

1 Introduction

In surveillance applications with different types of cameras, motion detection performs very important role in image processing. There are two salient features in motion detection system^[1]. One of them is region of interest and another one is region of awareness. The pivotal job of motion detection is to scrutiny a region of interest comprehended in a region of awareness. The region of awareness can be defined through camera geometry which is also significant to a location^[2]. Motion detection technology is not only used in security surveillance system, human tracking^[3], pose estimation^[4], reorganization^[5], but also in detecting fast moving vehicles and physiological applications^[6] such as detection of respiration, cardiac rhythms, knee osteoarthritis^[7] and blood pressure. Human heart has different idiosyncrasies, for example effects of ischemia in heart motion. For this type of cases^[8] proposed an LV motion detection that can differentiate between ischemic and normal heart condition. This method is also used to perform the nature of pathology.

The key idea of motion detection can be described. An image grabber grabs the images when a region of the image is changed in a certain level. In other word, the image is captured only when the pixels of a portion of the image exceeds a threshold according to system requirement^[1]. To illustrate the basic concept, the idea of achieving foreground mask can be described using threshold. Thresholding is used to binarize different images. It performs operation between two consecutive frames. For example,

we consider two frames - p_{th} and the following $(p-1)_{th}$ that are stand for $\mathcal{I}mg_m$ and $\mathcal{I}mg_{m-1}$ respectively. Then absolute difference image \mathcal{R}_m is calculated from (1).

$$\mathcal{R}_m = |\mathcal{I}mg_m - \mathcal{I}mg_{m-1}| \quad (1)$$

The resultant foreground mask $\mathcal{I}mg'_{(m-1,m)}$ is ordered in correspond to a threshold (Th). Different methods are proposed to figure out threshold value in order to binarize image difference^[9,10]. If a pixel value exceeds certain threshold, foreground frame can be selected. On contrast, it can be preferred to background. The definition of $\mathcal{I}mg'_{(m-1,m)}$ is given by (2)^[10].

$$\mathcal{I}mg'_{(m-1,m)} = \begin{cases} 1 & \mathcal{R}_m \\ 0 & \text{alternatively} \end{cases} \quad (2)$$

Then do morphology to $\mathcal{I}mg'_{(m-1,m)}$ so as to resist the disturbance of the background. Morphology^[11] is the process of converting gray scale image to binary image in which the image identification each pixel is restricted to a value of either 0 or 1.

There are a number of major issues that are related to motion detection such detection algorithm, image grabbing technique and overall running time. From all of these, motion detection algorithm is the most critical job of total procedure. A portion of section 2 is evaluated by three popular motion detection algorithms because motion detection has a great implication in industry, production and medical science. In spite of this, lots of shortcomings are seen in algorithms like entropy^[12], optical flow^[13,14] and researchers are making efforts to overcome constraints and find out the optimal solutions.

The challenge of motion detection system lies in secu-

*Corresponding author: S. Zahurul Islam

☎: +60 14 329 9692

✉: zahurul2003_du@yahoo.com

rity as secured system can be deployed in unmanned aircraft application. In this paper, an attempt is taken to improve security of the system for motion detection by suggesting an improved UNIX based method. The indispensable configuration procedure has also been described with appropriate command. Additionally, script for remote image files for micro-blogging application has been depicted with pseudo code.

2 Previous Research

According to [14], two types of motion detection are defined. These are region-based and pixel-based algorithms. Both types are widely used in motion detection although pixel-based type is often used in real time application. Huimin et al.^[15] have proposed imitating uni-Gaussian and mixed thresholding methods to discriminate practical and proposed methods. The proposed algorithm has been experimented very well in case of very slow moving objects where as fast moving objects have been ignored. One of the major critical issues in fast moving object motion detection in real time is acquiring accurate current background image. Showing a traffic surveillance system, Ji et al.^[16] have solved by color image difference model.

Genetic algorithm is very useful in case of optimization problems. Since the algorithm runs on the basis of iteration steps, the complexity of the algorithm is exponential. Using genetic algorithm Bevilacqua^[17] has shown motion detection system in terms of traffic monitoring application. In the experiment, it has been shown a good parameter optimization process albeit the complexity of the proposed algorithm has been totally ignored.

Three categories have been described below for motion detection divided into three parts. Researches^[15–18] have been going on these categories.

2.1 Background Subtraction Modeling

Background modeling is mostly used in moving objects detection^[19–21]. It means segmentation of foreground objects from dynamic scenes. The base of this model is background subtraction and the pivotal point is background creation. The concept of this model can be defined like this^[19] each point (n) of background is assumed as mean color and denoted as ∂_n and σ_n^2 as the covariance of that point. Thus (∂_n, σ_n^2) is stored as a color background. For each new frame f , $P_n(f)$ is the current color of pixel n . Then the background model is updated like pseudo code.

```

1 for (0 < γ < 1)
2 {
3   do if (n ∈ background pixel)
4   {
5     ∂n(f + 1) = (1 - γ)∂n(f) + γPn(f + 1);
6     σn2(f + 1) = (1 - γ)σn2(f) + γ(Pn(f + 1) - ∂n(f + 1))2;
7   }
8   else
9   {
10    ∂n(f + 1) = ∂n(f); % foreground pixel
11    σn2(f + 1) = σn2(f);
12  }
13 }
    
```

Lots of algorithms proposed in literature for ages presenting some common characteristics. Spagnolo et al.^[22] proposed multilayer approach for small changes in the background where object is moving at foreground loca-

tion. One of the major problems in background modeling is illumination changing in the environment. Huang et al.^[20] presented a modified approach by using background training parameter into Gaussian Mixture to improve the adaptive background model.

2.2 Optical Flow Model

An optical flow is defined by patterns of motion objects, edges, surfaces in an aesthetic scene brought on by relative motion between the scenario and a camera^[23]. A roll-out of optical flow on gradient is provided by Fleet and Weiss^[24]. To describe more precisely of optical flow, it is first elaborated maximum one pixel displacement or its large movements. The optical flow for a pixel (i, j) is given for two successive images k and $k + 1$. Then the movements of (i, j) denoted as $m_k(i, j)$ has become like (3) and (4).

$$m_k(i, j) = (x, y) \tag{3}$$

$$|I_k(i, j) - I_{(k+1)}(i + x, j + 1)| \tag{4}$$

Equation (4) is minimum when $-1 \leq x \leq 1$ and $-1 \leq y \leq 1$.

Brightness constancy is an inevitable part in optical flow technique^[1,24]. Brightness constancy is presumed by most researchers and sometimes disrupted by its parent nature. That is why; the output of optical flow field is neglected for approximation to two dimensional motion image. Fig. 1, Fig. 2 and Fig. 3 have been depicted here for better understanding of optical flow Lambertian sphere. Lambertian reflection model has been shown in Fig. 1. The corresponding equations (5) and (6) have been derived from Fig. 1. In both equations, $L(P)$ represents scene radiance, k is source brightness, γ' is surface reflectance and γ is effective source brightness^[25].

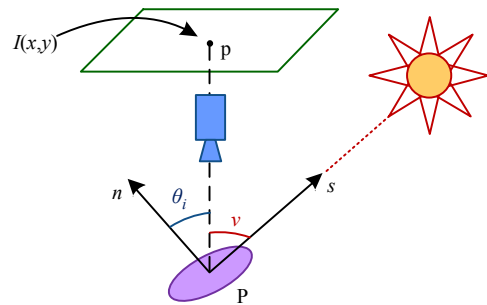


Fig. 1. Lambertian reflectance model^[25]

$$L(P) = \frac{\gamma'}{\pi} k \cos(\theta_i) \tag{5}$$

$$= \frac{\gamma'}{\pi} k (n \times s)$$

$$L(P) = \gamma (n \times s) \tag{6}$$

Fig. 2 shows a Lambertian sphere that rotates concentrating to a motionless light source and generates static image. In Fig. 3, intensities of range and optical image Fig. 3(a) and their first derivatives Fig. 3(b) have been shown. Based on Cross-Section of Lambertian Sphere X-Axis is Range, Y-Axis is Optical. The Point Light Source (PLS) is moving from 180° through 0 to -180°, and then looping back.

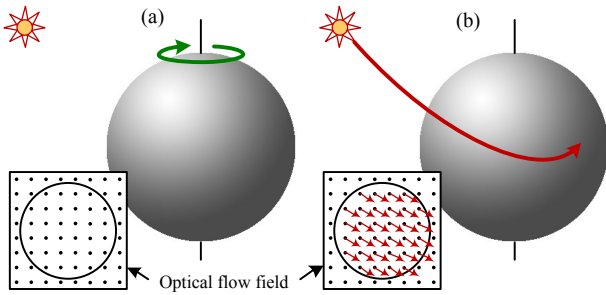


Fig. 2. A rotating Lambertian sphere with light source^[26,27]

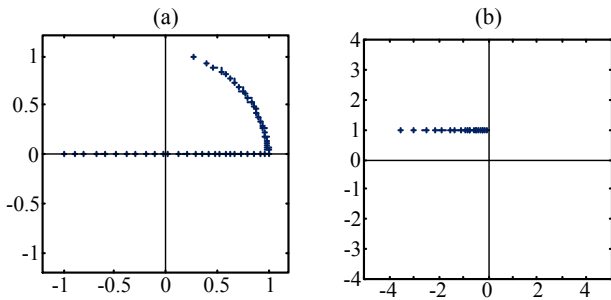


Fig. 3. Predicted histogram of (a) intensity of pixels of range and optical images; (b) first derivatives histogram output^[25]

2.3 Gradient-Based Optical Flow

There are some approaches in optical flow in which gradient technique is the most renowned^[14,24,28] one. The common idea of gradient-based optical flow is to estimation pixel intensities. It is assumed $f(x)$ be a one dimensional grey level function for spatial position. This is decoded by d between $time_1$ and $time_2$. Then generated shifted signal is expressed in (7). If it is applied Taylor expansion to (7), we can get like (8). After that, the difference between two signals of $f_{t_2}(x)$ and $f_{t_1}(x)$ is shown through (9). Finally, (10) presents first-order approximation with displacement which is denoted as \hat{d} . In case of a linear signal (Fig. 4(a)), displacement is the difference in signal values at a point divided by the slope. For nonlinear signals (Fig. 4(b)), the difference divided by the slope gives an approximation to the displacement^[24].

$$f_{t_2} = f_{t_1}(x - \hat{d}) \tag{7}$$

$$f_{t_1}(x - \hat{d}) = f_{t_1}(x) - d f_{t_1}'(x) + O(d^2 f_{t_1}'') \tag{8}$$

$$f_{t_2}(x) - f_{t_1}(x) = -d f_{t_1}'(x) + O(d^2 f_{t_1}'') \tag{9}$$

$$\hat{d} = \frac{f_{t_1}(x) - f_{t_2}(x)}{f_{t_1}'(x)} \tag{10}$$

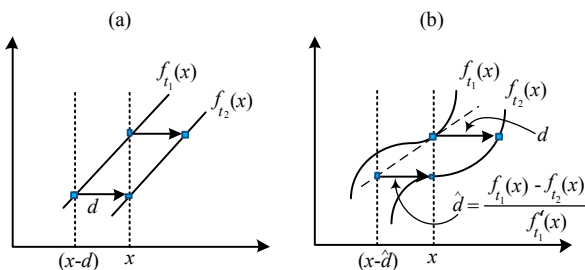


Fig. 4. Displacement (\hat{d}) of (a) linear and (b) non-linear signal^[24]

3 Method for Motion Detection

We propose a novel method that follows both temporal change and optical flow detection implemented in UNIX system. The proposed method has been implemented in Linux operating system. The most important of this program is configuration file. The details description of the configuration file is discussed in this section. Before going to configuration file, step wise diagram of this method has been shown in 5.

The method is described in the following steps.

- Step 1: Compute optical flow vectors for the highest level of the tree L (say we have a tree of L levels).
- Step 2: The vector values are made double.
- Step 3: First approximation is- optical flow vectors for the $(2i, 2j)$, $(2i + 1, 2j)$, $(2i, 2j + 1)$ and $(2i + 1, 2j + 1)$ pixels in the $L - 1$ level are assigned the value of the optical flow vector for the (i, j) pixel from the L level.
- Step 4: Adjust of the vectors of the $L - 1$ level in the tree. The adjustment method is detection of maximum one pixel displacements around the initially approximated position.
- Step 5: Smooth the optical flow field using Gaussian filter.

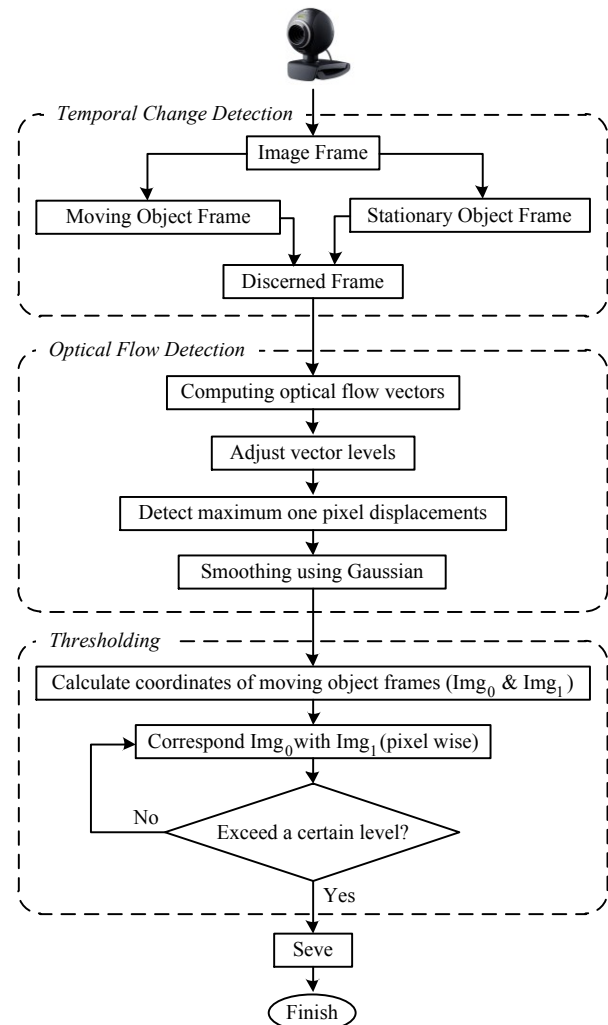


Fig. 5. Diagram of motion detection steps

Table 1. Some key commands for installing Motion

Serial	Command	Action
cmd1	tar-xvzf/path/to/motion-3.2.X.tar.gz; /path/to is the any directory	Untar the file -motion-3.2.X.tar.gz; /path/to is the any directory
cmd2	./configure	To setup for building environment for the C-compiler
cmd3	make	To build the code
cmd4	make install	To install the code
cmd5	make uninstall	To uninstall

4 Experiment and Results

In this experiment, it has been used Ubuntu which is a Debian-based GNU/Linux distribution operating system^[29]. To start the process of detecting motion, it is necessary to create an environment for running the program. The steps for configuration motion from sources are described in this section.

4.1 Configuration

Before starting the motion, it must be configured the motion.conf file according to the system capability, requirements etc. Table 1 describes necessary commands for Ubuntu system.

Besides these, there are copious options and all these need to be configured. Table 2 shows some of the options that we created according to our needs.

4.2 Alert Message

In Table 2, the option “on_picture_save” is used for uploading motion detected pictures to any social networking and micro-blogging service (Ex. Twitter is used) application by writing a scripting code. The pseudo code of the script for twitter is given as follow.

```

1 $verbose = 0;
2 $username <- 'user name of the Twitter; not email
   address'
3 $password <- 'Twitter Password'
4 $uploadAndPostSite <- "url (Uniform Resource
   Locator) of TwitPic API"
5 if (!$picture || !$username || !$password)
   then Error;
6 if ($verbose)
7 check username and password;
8 $response; use World Wide Web (WWW) agent to
   construct an object LWP::Protocol classes
   with arguments keep_alive and timeout;
9 $response = dispatch request with verifying
   security on behalf a user.

```

4.3 Results

The experimental results have been shown in Fig. 6 and Fig. 7. Fig. 6(a) displays the 90° rotated angel view of motion picture. Fig. 6(b) displaying the information while it the motion based picture is captured. Fig. 6(c) depicts the moving pixels of motion image. Finally, Fig. 6(d) unveils the pixel changing information of motion image.

On the other hand, Fig. 7(m-x) describes experimental result of motion detection of an object. The result explains that the camera can grab the object picture when it estimates any changes in both foreground and back-

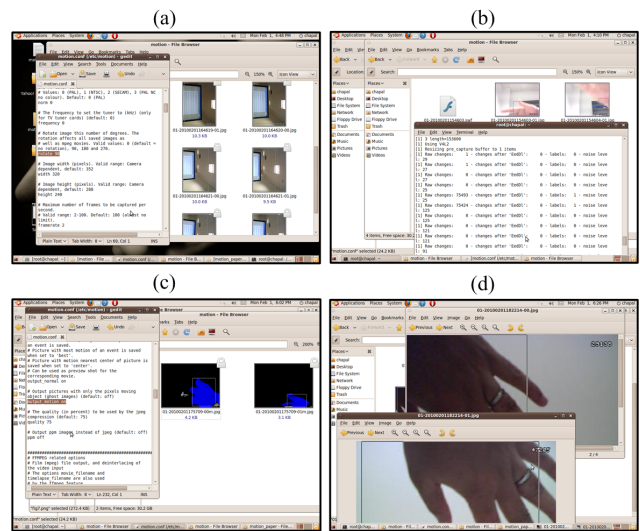


Fig. 6. Motion pictures in various configuration (a) rotate 90°; (b) setup mode on; (c) moving pixel; (d) pixel change display ground location.

Comparing to the conventional system, this UNIX system ensures the security level for motion detection^[30]. This will retain the intruders from hacking. However, one of the shortcomings can be notified in Fig. 7(q-r) that the program cannot detect any pixel change information when the object is reflected in a mirror.

Table 2. Relevant Configuration of motion.conf file

Options	Description	Configuration
cmd1	tar-xvzf/path/to/motion-3.2.X.tar.gz	Untar the file -motion-3.2.X.tar.gz; /path/to is the any directory
setup_mode	If it is on, motion starts in non-daemon (non-background) mode and gives details of changing pixel level (Fig. 6(b)).	;setup_mode on
rotate	Rotate image the given number of degrees (Fig. 6(a)).	; rotate 0 // 90 portrait
threshold	It is for the number of changed pixels that trigger motion.	;threshold 1500
output_motion	Generates the special motion type for moving object where the pixels changes (Fig. 6(d)).	;output_motion on
on_picture_save	Command to be executed when an image is saved.	;on_picture_save path name of the script file for uploading Twitter
on_motion_detected	Command to be executed when a motion frame is detected.	;on_motion_detected script for sending alert message through mail
thread	Required for using multiple cameras.	;thread /path/thread1.conf; thread /path/thread2.conf

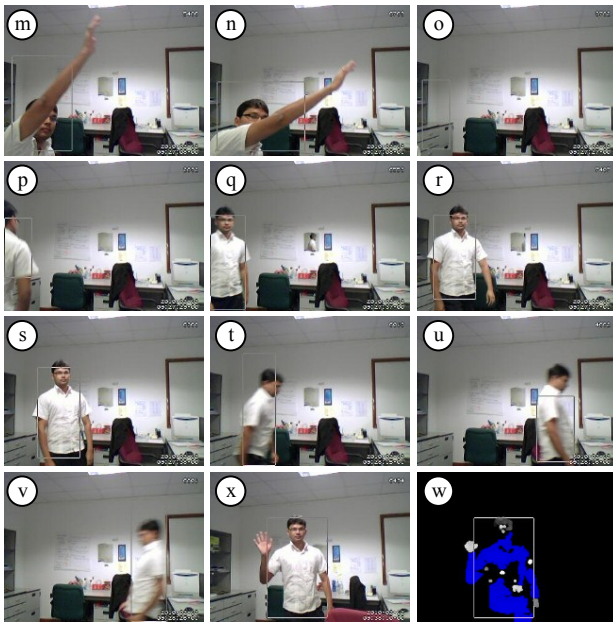


Fig. 7. Experimental results of detecting motion in different poses

5 Conclusion

In this work, a comprehensible motion detection method has been described with necessary tasks. Simply this motion detection system can grab images only when any object passes. So this is applicable for high speed vehicles to detect its speed, edges etc. choosing opt threshold and to trigger remote main server. Also the system can be convenient for counting number of vehicles in traffic signal point and set alarm to control room. However, enhanced research is needed on smoothness and choosing opt threshold. One promising area for future research can be applied different filter instead of Gaussian in case of smoothing. Overall, this UNIX based system is quite secured from hacking.

REFERENCES

- [1] L. Fang, Z. Meng, C. Chen, and Q. Hui, "Smart motion detection surveillance system," in *Education Technology and Computer, 2009. ICETC'09. International Conference on*. IEEE, 2009, pp. 171–175.
- [2] S. Gächter and T. Pajdla, "Motion detection as an application for the omnidirectional camera," *Research Reports of CMP, Czech Technical University in Prague, Omnidirectional Visual System (7)*, pp. 5–13, 2001.
- [3] R. Rangaswami, Z. Dimitrijevic, K. Kakligian, E. Chang, and Y.-F. Wang, "The sfinx video surveillance system," in *IEEE Conference on Multimedia and Expo*, 2004.
- [4] R. Melli, A. Prati, R. Cucchiara, L. de Cock, and N. Traficon, "Predictive and probabilistic tracking to detect stopped vehicles," in *WACV/MOTION*, 2005, pp. 388–393.
- [5] T. V. Duong, H. H. Bui, D. Q. Phung, and S. Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-markov model," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 838–845.
- [6] L. D. Canady, K. S. Honeyager, J. A. Helffrich, and D. A. Tong, "Motion detection for physiological applications," Nov. 12 2002, uS Patent 6,480,111.
- [7] S. Zahurul, S. Zahidul, and R. Jidin, "An adept edge detection algorithm for human knee osteoarthritis images," in *Signal Acquisition and Processing, 2010. ICSAP'10. International Conference on*. IEEE, 2010, pp. 375–379.
- [8] S. Lončarić and T. Macan, "Point-constrained optical flow for lv motion detection," in *Medical Imaging 2000*. International Society for Optics and Photonics, 2000, pp. 521–529.
- [9] Y. Liu and S. N. Srihari, "Document image binarization based on texture features," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 5, pp. 540–544, 1997.
- [10] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital image processing using MATLAB*. Gatesmark Publishing Knoxville, 2009, vol. 2.
- [11] S. W. Smith *et al.*, "The scientist and engineer's guide to digital signal processing," 1997.
- [12] Y.-F. Ma and H. Zhang, "Detecting motion object by spatio-temporal entropy," in *ICME*, 2001.
- [13] J.-E. Rivo and E. Cajote, "Object motion detection using optical flow," in *Digital Signal Processing Laboratory, Department of Electrical and Electronics Laboratory, University of the Philippines*, pp. 1–2.
- [14] M. Mosharraf, H. Sarker, K. Bechkoum, and K. Islam, "Optical flow for large motion using gradient technique," *Serbian Journal of Electrical Engineering*, vol. 3, no. 1, pp. 103–113, 2006.
- [15] W. Huimin, Z. Xiaoshi, Z. Yanling, and L. Na, "A new thresholding method applied to motion detection," in *Computational Intelligence and Industrial Application, 2008. PACIIA'08. Pacific-Asia Workshop on*, vol. 1. IEEE, 2008, pp. 119–122.
- [16] X. Ji, Z. Wei, and Y. Feng, "Effective vehicle detection technique for traffic surveillance systems," *Journal of Visual Communication and Image Representation*, vol. 17, no. 3, pp. 647–658, 2006.
- [17] A. Bevilacqua, "Optimizing parameters of a motion detection system by means of a distributed genetic algorithm," *Image and Vision Computing*, vol. 23, no. 9, pp. 815–829, 2005.
- [18] R. Cutler and L. Davis, "Real-time periodic motion detection, analysis, and applications," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on*, vol. 2. IEEE, 1999.
- [19] H. Liu, W. Pi, and H. Zha, "Motion detection for multiple moving targets by using an omnidirectional camera," in *Robotics, Intelligent Systems and Signal Processing, 2003. Proceedings. 2003 IEEE International Conference on*, vol. 1. IEEE, 2003, pp. 422–426.
- [20] T. Huang, J. Qiu, T. Sakayori, S. Goto, and T. Ikenaga, "Motion detection based on background modeling and performance analysis for outdoor surveillance," in *Computer Modeling and Simulation, 2009. ICCMS'09. International Conference on*. IEEE, 2009, pp. 38–42.
- [21] J. Wu and M. Trivedi, "Performance characterization for gaussian mixture model based motion detection algorithms," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 1. IEEE, 2005, pp. I–1097.
- [22] P. Spagnolo, M. Leo, T. D'Orazio, N. Mosca, and M. Nitti, "A background modelling algorithm for motion detection," *Proceedings of ISCCSP*, pp. 13–15, 2006.
- [23] D. DeCarlo and D. Metaxas, "The integration of optical flow and deformable models with applications to human face shape and motion estimation," in *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*. IEEE, 1996, pp. 231–238.
- [24] N. Paragios, Y. Chen, and O. D. Faugeras, *Handbook of mathematical models in computer vision*. Springer Publishing Company, Incorporated, 2010.
- [25] P. D. De Meer, *3-D Computer Vision CSc 83029, Photometric Stereo & Shape from Shading, Slide presentation by Ioannis Stamos*.
- [26] B. Jahne, *Computer Vision and Applications: A Guide for Students and Practitioners, Concise Edition*. Elsevier Science, 2000.
- [27] D. J. Fleet and A. D. Jepson, "Optical flow estimation," 2005. [Online]. Available: <http://www.cs.toronto.edu/~fleet/courses/2503/fall11/Handouts/opticalFlow.pdf>
- [28] J. S. Zelek, "Bayesian real-time optical flow," in *Vision Interface*. Citeseer, 2002, pp. 266–273.
- [29] "Ubuntu," [Online; accessed February-2013]. [Online]. Available: <http://www.ubuntu.com/products/whatisubuntu>
- [30] J. Viega and J. Voas, "The pros and cons of unix and windows security policies," *IT Professional*, vol. 2, no. 5, pp. 40–47, 2000.