Manufacturing Science and Technology 2(4): 67-74, 2014 http://www.hrpub.org
DOI: 10.13189/mst.2014.020401

Techniques on Video Streaming for Back Pain Prevention

N.S.A. Ramdan[1,*], A.Y. Bani Hashim[1], Seri Rahayu Kamat[1], Siti Azirah Asmai[2]

[1]Faculty of Manufacturing Engineering, Universiti Teknikal Malaysia Melaka, 76100 Durian Tunggal, Melaka, Malaysia
[2]Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, 76100 Durian Tunggal, Melaka, Malaysia
*Corresponding Author: sufiah.akmala@gmail.com

Abstract This paper is about the study of techniques on multiple video streaming, which is going to be used to prevent back pain among manual workers. The method used in this study is by using multiple video streaming which consists of three important elements; webcams , SimpleCV open-source and Python language. There are three techniques to do the video analysis. First is by using "on-the-shelf" software, the second is by combining the "on-the-shelf" software and self-programming, and the third one is by developing the "in-house" programming. The pilot test was held in Manufacturing Engineering Faculty, Universiti Teknikal Malaysia Melaka. The aim of the pilot test is to know whether the system is successful or not and to upgrade the system if there are any weaknesses.

Keywords Multiple Video Streaming, SimpleCV, Python

1. Introduction

Back pain is the common effect happened to manual workers, especially in a production line in the factory. The economic cost of back pain in the United Kingdom, for the year 2013, stated that the cost of low back pain, because of workdays lost amounts to £12.3 billion, which are equivalent to 1.5% of UK GDP and 22% of UK health care expenditure [1]. From this critical condition, quick action must be taken to prevent any more losses. Prevention is very important to reduce the number of back pain patient. From previous studies, the way of knowing people who had suffered back pain is only through surveys, but no way of preventing it [2]. In order to reduce and to prevent this back pain problem among manual workers, the author has come out with an advance idea to do the monitoring process, which is by using multiple video streaming techniques. This technique is first in the world up to our knowledge for monitoring back pain purpose.

2. Methodology

Based on Figure 1, to complete the system for the multiple video streaming, the video recording must first be analyzed. After that, the system must able to generate reports to be read by the person responsible for the monitoring such as production line supervisor. This is the basic concept of the monitoring system. There are various ways to do the video analysis. In this paper, we will discuss three techniques. First, by using "on-the-shelf" software, the second is by combining the "on-the-shelf" software and self-programming, and the third one is by developing own programming.

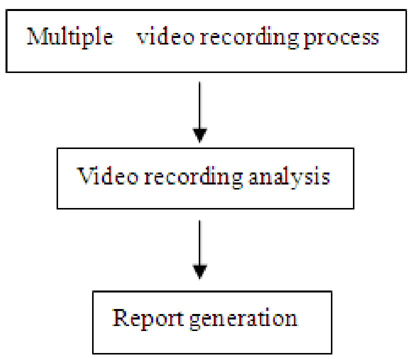

Figure 1. Basic concept of the multiple video streaming process.

2.1. On-the-shelf Software

On-the-shelf software is the software that is hundred percent ready made by the software developers. It is available on the internet. This software can be chosen and was downloaded from websites such as www.SecureCam.com, www.nchsoftware.com, . Based on Figure 2, for this technique, first, it will start with the process of finding a suitable multiple webcam software that can detect and record motions, generate and deliver reports through e-mails. It should be noted that not all on-the-shelf software has all the criteria that is needed and not all of the software is free.

If the installation is successful, the software is ready to be tested. Then the software is used to initialize all the cameras. If available, make sure the software is set to “record when motion is detected”. If the software is able to do that, then the next step can be continued. If not, the software will not help to achieve the objective of the system. Next, if the software is available to record when motion is detected, the software must able to send reports via SMS (short message service) or e-mail when the motion is recorded. If the software is able to send the report, the software can be selected, if not, the user must go back to the first step, which is to find the suitable multiple webcam software. In real experiment, the author has not yet found the exact software that fulfills all the criteria that will achieve the objective of the system to prevent back pain. Example of on-the-shelf software is SecureCam Software, EyeLine Video Surveillance Software, Debut Video Capture System, Xeoma Software [3][4][5]. Figure 3 shows an example of the experiments by using SecureCam software.

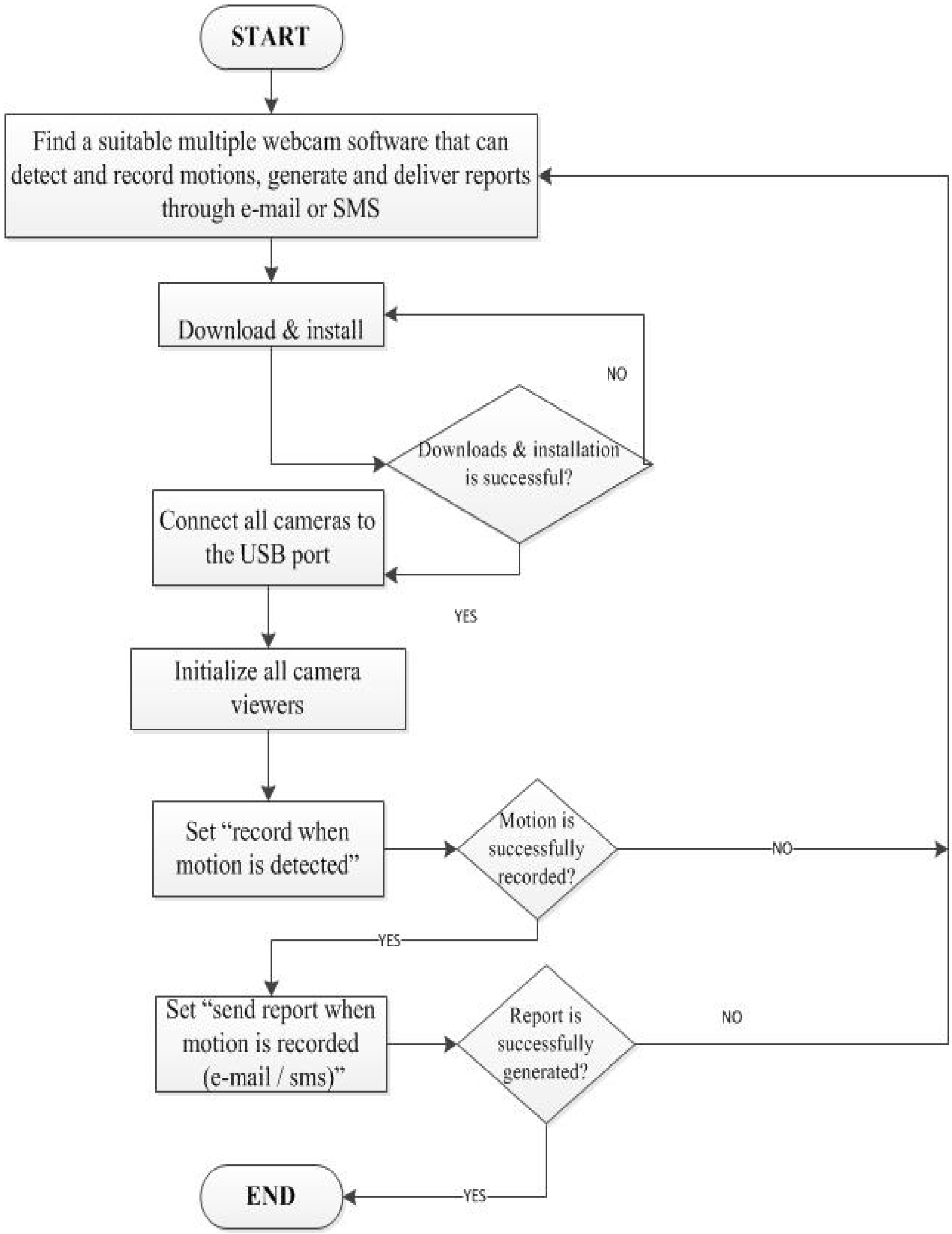

Figure 2. The flowchart to use “on-the-shelf” software technique

Figure 3. The SecureCam software is one of the *on-the-shelf* software.

2.2. Self-programming

The next techniques is by creating own programming which means that the user must create his or her own programming by themself that fulfils the criteria for the system. Figure 4 shows the flowchart to do self-programming. First, the design of the system must be planned well.. Then, the framework and the language of the system are decided.

The complete system can be assessed and validate by doing the experiment for the first time. If the assessment and the validation are showing that the system is doing well, then the system can be accepted. If not, the development and modification of the system must be repeated again. Example of suitable framework is SimpleCV (Simple Computer Vision) and OpenCV (Open Source Computer vision) [6][7]. Both of these framework are using Python language [8]. Figure 5 shows one of the many vision system product from SimpleCV framework.

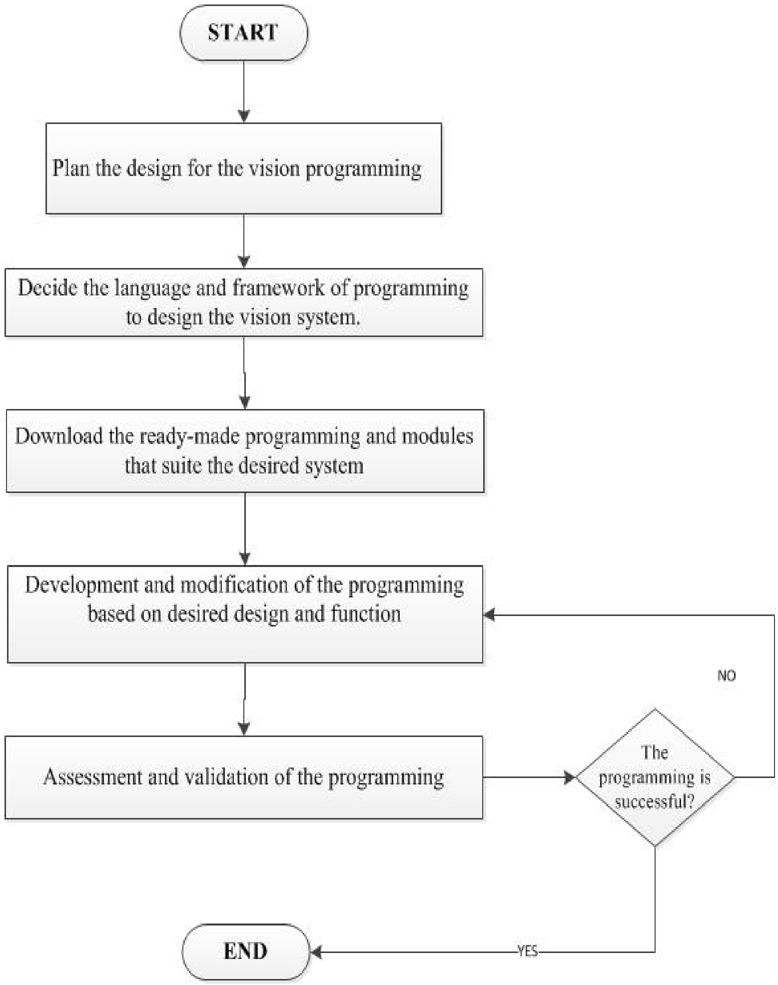

Figure 4. The flowchart to do self-programming.

Figure 5. One of the product from SimpleCV framework.

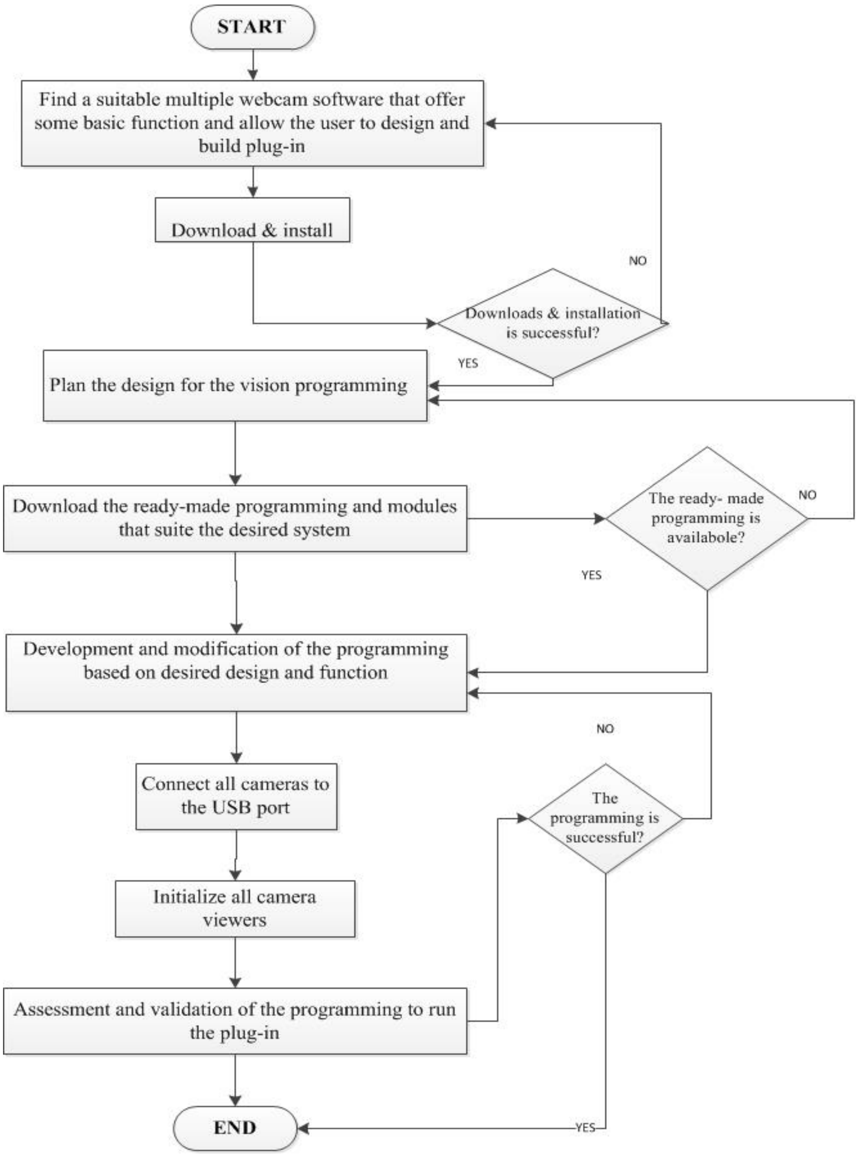

Figure 6. The flowchart of by combining the on-the-shelf software and self-programming method.

2.3. Combining the On-the-shelf Software and Self-programming for Video Streaming

Figure 6 shows the flowchart of method to do multiple video streaming by combining the on-the-shelf software and self-programming method. The first step to start this process is by finding the suitable multiple webcam software that offers some basic function such as motion detection, auto-motion recording, and report processing and report delivery. As for the reminder, not all software will have all the criteria that are stated before.

After finding the right multiple webcam software, download and installation may be started. Then, start using the software that has been downloaded and initialize all camera viewers. To do a complete and proper setting of the webcam viewers, the setting part must be explored completely. From that, choose "record when motion is detected" or any setting that similar to that. Also, choose the report delivery method, whether via SMS or e-mail.

Next, the development and modification can be done if the ready-made modules are not suitable with the desired system. Assessment and validation can be done after all the programming is done. An example of the software that can adapt with the technique is iSpy an open source camera security software [9]. iSpy software will allow the user to design their own plug-ins.

3. Video Processing Formula

3.1. Overview of Basic Properties

Digital video comprises a series of orthogonal bitmap digital images, which is displayed in rapid succession at a constant rate. These images are called frames.(ref). The measure of the rate at which frames are displayed is, frame per second (FPS) If it has a width of W pixels and a height of H pixels, so the frame size is WxH. A property of pixels is their color.

The color of a pixel is represented by a fixed number of bits. The more subtle the variations of colors can be produced, the more number of bits the pixel have. This is called the color depth (CD) of the video. An example video can have a duration (T) of 1 hour (3600sec), a frame size of 640x480 (WxH) at a color depth of 24bits and a frame rate of 25fps[10]. This example video has the following properties:

pixels per frame
= 640 * 480 = 307,200
bits per frame
= 307,200 * 24 = 7,372,800 = 7.37Mbits
bit rate (BR)
= 7.37 * 25 = 184.25Mbits/sec
video size (VS)
= 184Mbits/sec * 3600sec = 662,400Mbits = 82,800Mbytes = 82.8Gbytes

The most important properties are bit rate and video size. The formulas relating those two with all other properties are:

BR = W * H * CD * FPS
VS = BR * T = W * H * CD * FPS * T
(units are: BR in bit/s, W and H in pixels, CD in bits, VS in bits, T in seconds.

3.2. Normalization

In image processing, normalization is a process that changes the value of pixel intensity range. Normalization is sometimes called contrast sketching or histogram stretching. Normalization transforms an n-dimensional grayscale image as in Equation (1):

$$I: \{ X \subseteq R^n \} \longrightarrow \{Min, ..., Max\} \qquad (1)$$

with intensity values in the range (Min, Max), into a new image as in Equation (2):

$$I_N: \{ X \subseteq R^n \} \longrightarrow \{newMin, ..., newMax\} \qquad (2)$$

with intensity values in the range (newMin, newMax). The linear normalization of a grayscale digital image is performed according to the formula in Equation (3):

$$I_N = (I - Min)\frac{newMax\text{-}newMin}{Max\text{-}Min} + newMin \qquad (3)$$

For example, if the intensity range of the image is 50 to 180 and the desired range is 0 to 255 the process entails subtracting 50 from each of pixel intensity, making the range 0 to 130. Then each pixel intensity is multiplied by 255/130, making the range 0 to 255.

Normalization might also be nonlinear, this happens when there is not a linear relationship between I and IN. An example of non-linear normalization is that when the normalization follows a sigmoid function, in that case, the normalized image is computed according to the formula in Equation (4):

$$I_N = (I - Min)\frac{newMax\text{-}newMin}{Max\text{-}Min} + newMin \qquad (4)$$

Where α defines the width of the input intensity range, and β defines the intensity around which the range is centered [10].

4. Experimental Simulation

In this project, one of the most important steps is the simulation before continuing to the real experiment. This simulation is important because it will help the author to estimate many factors before starting the real experiment. This simulation also helps to show the movement of the whole body during the experiment For this simulation, the author had used the CATIA (Computer Aided Three-Dimensional Interactive Application) software[11].

From the Figure 7(a), we can see that the mannequin is used to simulate the experiment. Figure 7(b) shows three markers are put at the back of the human to differentiate between three parts of the backbone. These markers are important because the distance between the three markers will an angle difference. This will allow the cameras to detect the angle differences. Figure 7(c) shows the mannequin posture is starting to bend its body to the position as it wants to pick up something from the floor. Figure 7(d) shows the maximum bending that can be achieved by the mannequin that is 45 degrees [12]. It can be seen that the three markers have developed an angle different.

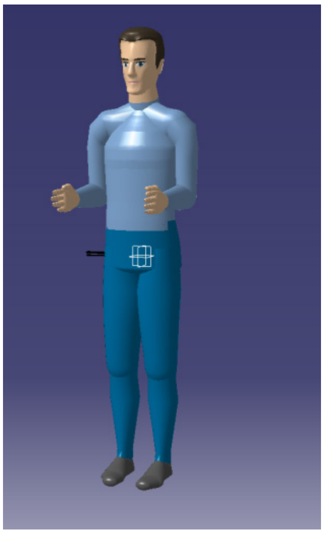

Figure 7(a). The manniquin

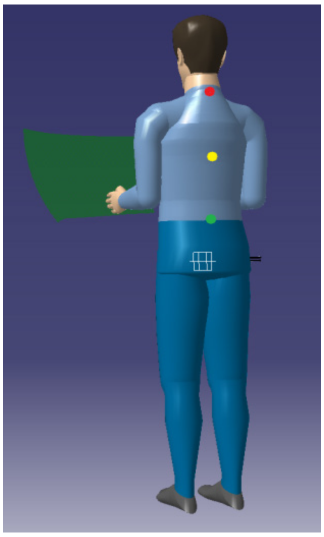

Figure 7(b). The back of the manniquin

Figure 7(c). The manniquin start to bend

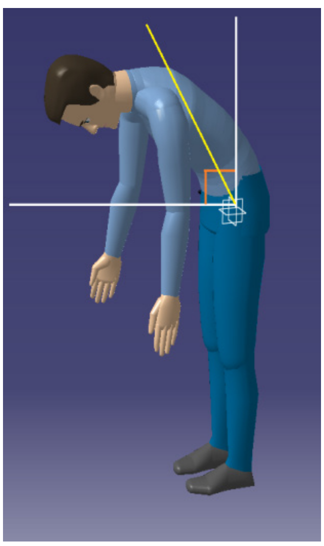

Figure 7(d). The back of the manniquin when it bends.

5. Pilot Test

The objective of this pilot study is to do a pilot test of the system in the laboratory. Before doing the experiment, the environmental factor must be considered first, such as the number of samples, light intensity, wind speed and temperature of the surrounding.

5.1. Subject

Subjects of respondents are the workers or technicians that use the machine in the laboratory. The demographic information of the respondent also must be noted such as age, height, weight, and workstation.

5.2. System Configuration

The system configuration for this vision monitoring system is comprised of cameras, central processing unit and the Python programming. Figure 8(a) shows three cameras are being used to do the recording. This is important because the camera must capture a 3D image of the worker and it must be multiple video streaming. Figure 8(b) shows the Central Processing Unit (CPU) where the cameras are connected to the computer. Figure 8(c) shows the Python programming that will do the image processing. connected to the computer. Figure 8(c) shows the Python programming that will do the image processing.

Figure 8 (a). The cameras

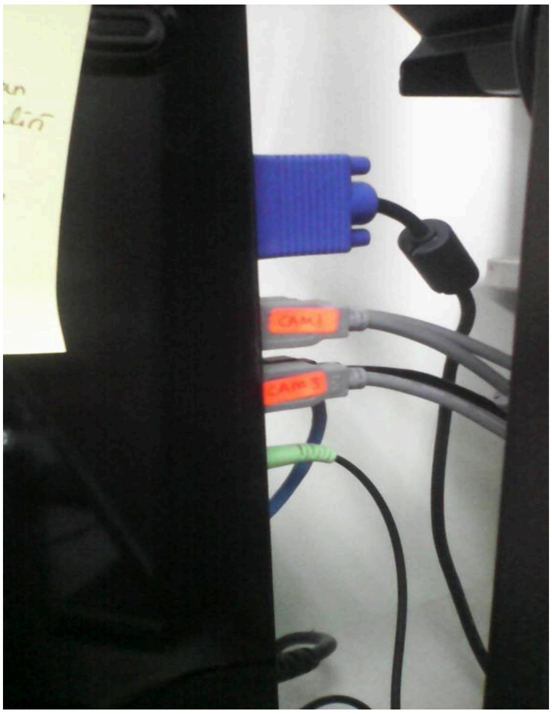

Figure 8 (b). The Central Processing Unit (CPU)

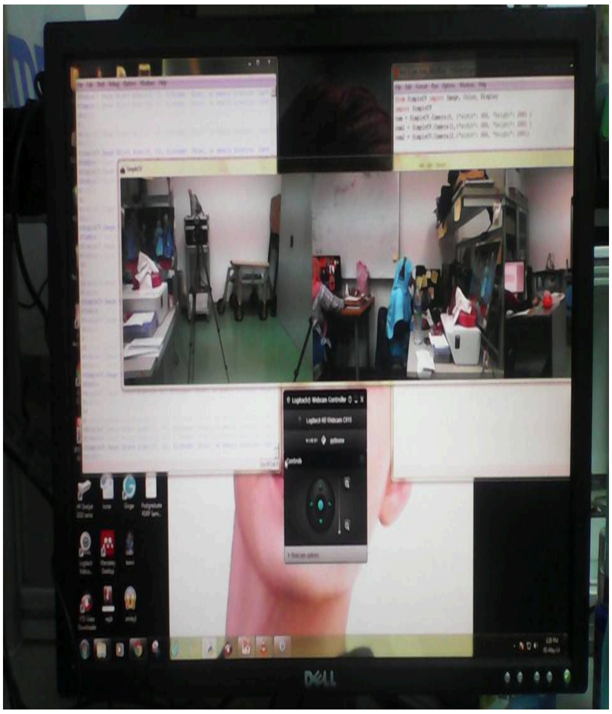

Figure 8 (c). The program

5.3. Experimental Equipment and Procedures

Figure 9 shows the experimental setup for the pilot test. The three cameras are arranged according to the height of the worker and the configuration workspace.

Figure 9. The experimental setup.

Equation 5 shows that, to measure light intensity, one Lux is equal to one lumen per square meter:

$$1 \ \text{lx} = 1 \ \text{lm/m}^2 = 1 \ \text{cd}\bullet\text{sr/m}^2 \tag{5}$$

The characteristics of the experimental surroundings that must be noted, are such as experimental place, temperature and light intensity. After that, when the recording is done, Python analysis must be made to the video recording. From the analysis, the result of the system will be known, whether the system can be used or not. The weaknesses of the system that need to be fixed also can be identified.

6. Summary

Three techniques can be used to develop a multiple video streaming system. First is by using “on-the-shelf” software, the second is by combining the “on-the-shelf” software and self-programming, and the third one is by developing the “in-house” programming or doing own programming. The most suitable technique is by developing own programming. This is because the technique is more flexible in order to achieve the objectives of the system. Even though it is a little hard at the beginning, but it will be more easier to be fixed and edited if something goes wrong on the code. Furthermore, if the user decided to use the SimpleCV or OpenCV open-source, it is more easier because the Python language that it uses is very user friendly and is used worldwide. The pilot study is still in progress and is aimed to finish on September 2014..

Acknowledgements

This research is supported by the Universiti Teknikal Malaysia Melaka, research grant identification number PJP/2013/FKP(5D)/S01173.

REFERENCES

[1] E. Union, “Societal Impact of Pain (SIP) Chronic pain in Europe pain as quality indicator,” no. May, 2013.

[2] A. Barkhordari, “The Prevalence of Low Back Pain among Nurses in Yazd, Southeast Iran,” Int. J. …, vol. 5, no. 1, pp. 19–21, 2013.

[3] “EyeLine Video Surveillance Software,” 2014. [Online]. Available: http://www.nchsoftware.com/surveillance/.

[4] “Debut Video Capture System,” 2014. [Online]. Available: http://www.nchsoftware.com/capture/.

[5] “Xeoma - best webcam software for Windows, Linux and Mac OS X,” 2014. [Online]. Available: http://felenasoft.com/xeoma/en/.

[6] “SimpleCV.” [Online]. Available: http://simplecv.org/.

[7] “OpenCV | OpenCV.” [Online]. Available: http://opencv.org/.

[8] Python, “Python.” [Online]. Available: http://www.python.org/. [Accessed: 13-Jan-2014].

[9] Sean Tearney, “ispy connect,” 2013. [Online]. Available: http://www.ispyconnect.com/. [Accessed: 20-Jan-2014].

[10] Wikipedia, “Digital Video-Wikipedia,” 2014. [Online]. Available: http://en.wikipedia.org/wiki/Digital_video. [Accessed: 26-Jun-2014].

[11] Wikipedia, “CATIA-Wikipedia,” 2014. [Online]. Available: http://en.wikipedia.org/wiki/CATIA. [Accessed: 26-Jun-2014].

[12] N. Hashim, S. R. Kamat, I. Halim, and M. S. Othman, “A study on push-pll analysis associated with awkward postre among workers in aerospace industry,” pp. 233–239, 2014.