

Multiple objects  
tracking via  
collaborative  
background  
subtraction computer  
scien...



**ASSIGN  
BUSTER**

Multiple Objects Tracking Via Collaborative Background Subtraction. Object tracking system is a group of integrated modern technology working together to achieve certain of purpose like monitoring, tracking moving object such as vehicle. The main purpose of the object tracking is to achieve monitoring purpose such surveillance in restricted area, providing information about moving vehicle located at road to Intelligent Traffic System and traffic monitoring. This project discusses the development of the object tracking system and the idea of this system is based on vision system available on current market. For this object tracking system, user can monitor and track moving object such as vehicle where the vision system been placed. Software MATLAB is used to program algorithm like detecting and tracking moving object where the vision system is placed and display moving object image for user.

## TABLE OF CONTENTS

Page TITLE

DECLARATION i

ABSTRACT ii

ABSTRAK iii

TABLE OF CONTENTS iv

LIST OF TABLES vii

LIST OF FIGURE viii

## LIST OF ABBREVIATIONS ix

## LIST OF TABLES

Table No. Description Page

Table 4. 1: Summarize of three experiment conduct previously. 17

## LIST OF FIGURES

Page

Figure 2. 1: Example of Median Filtering, value of current pixel will replaced using new median value 5

Figure 2. 2: Normal presentation of a straight line 7

Figure 3. 1: Relationship between webcam, MATLAB and GUI 11

Figure 3. 2: Flow of work for vehicle tracking system 12

Figure 4. 1: Logitech Quickcam Pro 4000 Image 14

Figure 4. 2: Image captured for YCbCr return color space 15

Figure 4. 3: Image captured for grayscale return color space 16

Figure 4. 4: Image captured for grayscale return color space 17

Figure 5. 1: Example of frame differencing 23

Figure 5. 2: Memory of cache been flush 24

Figure 5. 3: GUI window layout design 25

Figure 6. 1: Display figure when there is no moving object 28

Figure 6. 2: Moving curtain cause by wind 28

Figure 6. 3: Moving stand fan motion. Frame start from up to bottom and left to right. 31

## LIST OF ABBREVIATIONS

CCD Charge-couple Device

FPS Frames per Second

GUI Graphical User Interface

ID Identification Number

USB Universal Serial Bus

VGA Video Graphics Array

## CHAPTER 1 INTRODUCTION Overview

Object tracking system is defined as a real time vision system which is capable to perform desired surveillance task without human supervision (Nguyen, K. et al., 2002). Besides that, object tracking system is able to detect object which is moving in street such as vehicles or pedestrian without human assistance. Furthermore, an object tracking system may also send amount of vehicle which is moving in desire area to assist data

collection for Intelligent Transportation System (R. Reulke et al., 2002). This tracking system may also have the abilities to resist with environmental changes such as shadow of surrounding building or even slow moving vehicles. Therefore, a quick response for vision fields such as real time street monitoring system which are able to perform moving object detection. In this project, the main purpose is to design an object detection mechanism for an object tracking system, from connecting a vision system to a computer. The target is to build an applicable object tracking system.

Object tracking system can distinguish between static background and moving objects by itself and able to display and tracking moving objects if any moving objects detected. Hence, it allow us to monitor a heavy load street which having high volume of usage. Furthermore, it can contribute data collection if those areas contain Intelligent Traffic System which can reduce time of waiting for vehicle located at traffic light.

Since year 2000, plenty of fast response time or accurate object detection algorithm had been release such as background subtraction, mean shift, Kalmen filter, Markov Chain Monte Carlo, Kernel Density and others.

Object Tracking System consists of two major systems which are vision system and moving object detection and tracking software system. The vision system is responsible to export video stream captured and send to tracking system. Meanwhile, the tracking system is to let user monitor and been inform if moving object detected. In this project, object tracking system will be design and developed to ensure it is capable to detect and tracking moving object such as vehicles moving in street. Due to this, it could not

effectively detect fast moving object, surrounding light intensity is too low or shadow of building. As a result, the detection algorithm should fast enough to process each frame coming from vision system and can able to encounter problem stated before such as shadow surrounding and slow responding time by tracking system.

### Problem Statement

The current real time object tracking systems developed usually cannot eliminate having slow respond during tracking object which will limit the robustness of object tracking. Hence, the algorithm that able to having less computation time is necessary to be developed. Background subtraction at the initial detection will save computation time for faster response to detect an object in real time. To obtain more accurate tracking result, a more precise detection and tracking algorithm will be carried out. It is believe to track the moving object using this algorithm will taking less time and providing more accurate result.

### Objective

The aim of this project is to detect multiple moving objects through real time vision system. This project's aim can be realized by accomplishing the following sub-objectives.

To study and identify practical parameters to track a moving object.

To implement background subtraction for real time detection purpose.

To enhance the developed algorithm for continuous tracking purpose.

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

To ascertain and enhance performance of develop background subtraction based tracking system.

#### 1. 4 Scope of Work

The main scope of this project is to build an object tracking system capable of detect and track moving object. The object tracking system includes a vision system and an image processing system. The image processing system will able to detect moving objects and tracking it continuously.

MATLAB control m-file will be acts as core of the object tracking system, it will be use as detect and track moving vehicle in video supply by vision system. The vehicle tracking system will display in GUI window.

Vision system will be use as a supplier to supply tracking system that video capture in desire area. This system should be small enough so that it can be easily set up or take away.

#### Organization of the Report

This report includes seven chapters, each chapter is properly divided and plan. Vision system and object tracking system will be discussed in each chapter.

Chapter 2 discussed about review of object tracking and detecting method available nowadays.

Chapter 3 explains about flow of work require for this tracking system, parameter require during tracking system is running, input and output

prediction and concept how to build this tracking system using vision system available in market.

Chapter 4 explains hardware and software setup before this tracking system was starting to run. This is to ensure vision system will supply appropriate video require for tracking system and MATLAB will provide suitable arrangement such as memory to process the video supply by vision system.

Chapter 5 discuss about algorithm using in this project that is background subtraction using frame difference. In this chapter, an M-file will be constructing and including function requires establishing the tracking system. The tracking system should able running using hardware and software setup preparing at previous chapter together with this M-file.

Chapter 6 show image output and result obtain during this tracking system is running. Firstly it will show successful background subtraction and secondly it will show distortion of surrounding such as shadow of object.

Chapter 7 will summarizes and concludes the report by stating the limitations of the project as well as the future work of the project.

## CHAPTER 2 review of object tracking and detecting method

2. 1 Overview  
In this chapter, review of existing method to detect and track object will be discussed. Algorithm that suitable for detect and tracking also will be studied. Several algorithms will be review by student.

### 2. 2 Median Filter



Median Filter, use to reduce small noise in an image is a commonly used technique (Al-amri, S. S et al., 2010). According research by Boyle, small noise normally appears very distinct and it's having quite different value in grayscale within its neighbor pixel values. By changing its gray value to the median of neighboring pixel value, the noise can be eliminating using this technique.

Using example in Figure 2. 1, the value of neighboring pixels are 115, 119, 120, 123, 124, 125, 126, 127 and 150. By calculating median value using these neighbor pixels, we can obtain median value is 124. Replacing pixels in centre using median value will eliminate the noise.

Figure 2. 1: Example of Median Filtering, value of current pixel will be replaced using new median value

In order to obtain more accurate median value, we should increasing number of neighbor which involve in median value calculating. This technique will become more and more complex when dealing with bigger image. Besides that, computation cost and time require is relatively high because it needs to sort all value in neighbor.

### 2. 3 Canny Edge Detector

Canny introduce a well-known technique using edge detection (Neoh, H. S et al., 2005). This method requires few steps to track an object.

Remove small noise using smooth a image

Two gradient images are generated on both vertical and horizontal direction using one of the gradient operators based on previous image.

Result denoted as  $G_x(m, n)$  and  $G_y(m, n)$  where  $m$  and  $n$  are pixel coordinate.

Calculate edge magnitude and direction images from previous two images.

Edge magnitude,  $M(m, n) =$

Edge direction,

Threshold the edge magnitude image  $M(m, n)$ . Set pixel to zero if their value below a predefined threshold.

Reduce edge breadth by non-maxima operation on  $MT(m, n)$  the non-zero pixels in  $MT(m, n)$  are set to zero if their value are not greater than their neighbors along the direction indicate by.

Result is threshold using two identical thresholds:  $T_1$  and  $T_2$  where  $T_1$

Edge with a magnitude less than  $T_1$  will be removed and those greater than  $T_2$  are detect as real edge.

Edges with magnitude between  $T_1$  and  $T_2$  also detected as edges if they connect to an edge pixel.

## 2. 4 Hough Transform

This technique detects object whose shape can be parameterized in a Hough parameter space (Gurbuz, A. C. et al., 2008). These objects include

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

polynomials, straight line, circle and etc. The peaks detected in Hough parameter space is used to describe the object space.

An example, line segment can be described using a parametric notion:

Where  $r$  is length of a normal from origin to this line and  $\theta$  is orientation of  $r$  with respect to  $x$ -axis.

$\theta$

Figure 2. 2: Normal presentation of a straight line

Using this normal presentation, we can transform the points on the line to curve in a Hough parameter space whose coordinates represent the normal length and orientation. Points which are on the line generate curves intersecting at a common point  $(r, \theta)$ .

## 2. 5 CamShift

CamShift or “ Continuously Adaptive Mean Shift” track objects based their color. This technique was developed and detects an object using centre and size of the object in a given image (Ganoun, A. et al., 2006).

Step of tracking an object is as follows:

Set the size of search window.

Initialize location of searching window.

Location of centroid within search window based on the 0th and first moment been computed.

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

Search window is centered at the centroid.

Step three and step for is repeated until it has move for distance lee then a preset threshold.

In order to use this technique, an identical color of object must be use. Hence, one object with complex color is not suitable for this technique.

## 2. 6 Kalman Filter

This algorithm is a state estimation based on feedback control mechanism (Donald, J. S. et al., 1998). This filter will predict the process state and then obtains feedback from the measurement.

Equation for Kalman filter is divided to two groups:

Time update equation.

Measurement update equations

Time update equation is used to predict current state and error covariance. Output of these equations is a state of prediction for next time step. In the other hands, the measurement update equations incorporate a new measurement into their prior prediction. Output of this is an improved estimation compared to other estimation.

However, Kalman Filter cannot detect fast moving object such as moving vehicle in Highway, this is because changes in speed, acceleration can be dramatic during two consecutive frames.

The Kalman filter is not fast enough to respond to constant and sudden changes of system rate. Hence, it is not suitable for detection purpose which require less computation time.

## 2. 7 Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) is a class of algorithm for sampling from probability distributions based on constructing a Markov Chain that has desired distribution as its equilibrium distribution.

In order to construct a Markov chain Monte Carlo, it must contain three main stages (Jia, Y. Q. et al., 2009):

Model Construction.

Image is first pre-processed to retrieve its edge features. Models of roads and vehicle also been defined according for this method.

Bayesian formulation.

Since vehicle detection and segmentation problem is casted as Bayesian problem of finding a MAP solution, a corresponding formulations been defined. Prior probability and like hood of vehicles proposal are defined from which the form of the posterior probability is derived to evaluate different proposals.

Detect a vehicle using MCMC.

Construct a Markov chain to sample the proposal in the parameter space. Monte Carlo method with simulated annealing been used to search for the position and other related parameters that fixed actual vehicles most.

## 2. 8 Background Subtraction

In background subtraction, two image been captured in same location will be compared. Assume first image did not contain any moving object (empty background) and next image contain one moving object. Minus the second image with first image will contain moving object only since background of image been subtracted (Fukushima, H. et al., 1991).

The image is read as array format in the image processing, which each pixels is represented by matrix coordinates  $(x, y)$ . The intensity at position  $(x, y)$  is define by  $I(x, y)$ .

(4. 1)

From Equation 4. 1 Where  $I_c$ ,  $I_b$ ,  $I_s$  are the contributions from the foreground objects and background objects respectively. In the image for the subtraction, the brightness is written as

(4. 2)

The position adjustment between the two images is easily carried out by using the foreground objects. In order to obtain the foreground object, the first image is subtracted from the second one which contains the foreground object as show in Equation 4. 2.

## CHAPTER 3 CONCEPTUAL DESIGN 3. 1 Introduction

Method of how to detect and track object will discuss in this chapter. The vision system will capture video in a desire area and send that video to MATLAB for processing. The MATLAB will process data coming from vision system and performing tracking action.

Figure below show the mechanism for vision system and MATLAB. The vision system includes webcam which can connect to a computer using USB. The MATLAB will get data from vision system and processing the data. After that, A GUI window will show moving object if moving object exist capture by vision system.

Webcam

MATLAB

GUI

Figure 3. 1: Relationship between webcam, MATLAB and GUI

MATLAB been chosen as platform for detecting and tracking due to it contain powerful toolbox which can use to synchronize with webcam and can produce a simple detect and track vehicle tracking program. Besides that, it also can produce a GUI window which is requiring for the tracking system.

### 3. 2 Flow Chart of Work

In this section, flow of work requires detecting and tracking moving object will be further discussed. Frame differencing will be using to subtract the

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

background and obtain the masking of moving object. In order to obtain more accurate result, a more accurate algorithm will be use to track moving object.

Input Video Frame from camera

Pre-processing

Store the current frame as background

Subtract the next frame with background image

Save into memory

Update current frame as background

Display moving object and track it continuously.

Figure 3. 2: Flow of work for object tracking system3. 3 Discussion

In this chapter, draft and prototype of tracking system been discussed. In order to achieve this objective, the tracking system will be build based on conceptual design discussed previously.

In the following chapter, pre-processing will be elaborated and method to connect webcam with MATLAB will be show. Preparation configuration also will discuss in details.

CHAPTER 4Hardware and Software Setup4. 1 Overview



Hardware and software setup is defined as a preparation before a simulation is set up in either hardware (tools or instrument) or software (simulation program, programming language) by designer. A setup describes a system will be perfectly connecting between hardware and software to achieve certain mission. Engineer use a tools or instrument that either ready in market or design it according to their requirement. In other way, software such as scientific program also available in market, all that engineer need to do is fully utilize the program by design an efficient flow which can achieve their expectation. Engineer can develop a surveillance system and by using a mathematical modeling to analyze and obtain object which is moving from view of camera.

In this chapter, hardware and software setup is carried out for the design of a street monitoring system. It includes the connecting webcam to MATLAB which will let MATLAB ready to get real time video recording from webcam, M-file coding which contain algorithm to extract background which is static from object (vehicles or pedestrians) which is moving. Lastly, is to show image which is moving after process of background subtraction been executed in form of GUI.

#### 4. 2 Tools and Software

In this section, tools and software using along this project will be describe in details of how they contribute in this project. Tools using in this project is a webcam which can connect to computer via USB 2. 0 connection, it can either capture a static picture or even recording a video which can be treat as real time recording device. Software using in this project is MATLAB

R2009a. In MATLAB R2009a, toolbox which will be use to develop this street surveillance system is Image Acquisition Toolbox and Image Processing Toolbox. Image Acquisition Toolbox will be use to establish a real time recording from webcam and delivered to MATLAB. In other hands, Image Processing Toolbox will be use to process continuous frames capture which is stored in MATLAB and show moving object which is process by using background subtraction.

#### 4. 2. 1 Webcam

In this project, student will use Webcam which is product of Logitech with model Logitech Quick Cam Pro 4000.

Figure 4. 1: Logitech Quickcam Pro 4000 Image

Source: Logitech Software Support (2010)

Logitech Quick Cam Pro is a webcam that able to capture video in 640 x 480 resolutions and able to snapshot a picture with 1280 x 960 resolutions.

Besides that, it also contains a build in microphone which able to record sounds around that webcam been located and activated. Video capture from this webcam is using advance VGA CCD sensor and up to 30 fps. (Logitech, 2004)

In order to try different video input format, student tried several video input format available for this vision system such as YCbCr, grayscale and RGB. These three return color space been chosen due to vision system using at here, Logitech Quick Cam Pro 4000 only support these three return color space.

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

Three experiments will be performing to choose the suitable return color space from YCbCr, grayscale and RGB. In each experiment, three cases will be using to test different light intensity towards an object (battery) that is low, normal and high.

For low light intensity, surrounding of image captured should be dark enough. Normal light intensity test will be performing at inner space with medium light intensity and camera should not point toward a direction with strong light source such as sun or spotlight. In the last case, camera will be capture image in direction towards strong light source such as torchlight.

These experiments will be tested using webcam connect to MATLAB and executing command codes. Summarize of three experiment will be included in Table 4. 1.

Experiment 1: Using YCbCr as video input format and display as figure.

After webcam is connecting to MATLAB, code as below will be executing to perform the test.

```
vid = videoinput('winvideo', 1);  
  
set(vid, 'ReturnedColorSpace', 'YCbCr');  
  
preview(vid)
```

From Figure 4. 2(a), image obtained almost in dark due to low intensity of light surrounding object. Image can be seeing using human eyes in clear view for Figure 4. 2(b). For last case, object still can consider as clear

although white spot cause by strong light source located at upside of Figure 4. 2(c).

(a) (b) (c)

Figure 4. 2: Image captured for YCbCr return color space(a) Low light intensity (b) Normal light intensity (c) High light intensity

From this experiment, this return color space is potential to be used in this project. It does not lose color property and only having small changes of color during in high light intensity situation.

Experiment 2: Using grayscale as video input format and display as figure.

To perform this experiment, previous video object should delete from MATLAB workspace and executing following code.

```
vid = videoinput('winvideo', 1);  
  
set(vid, 'ReturnedColorSpace', 'YCbCr');  
  
preview(vid)
```

From both Figure 4. 3(b) and Figure 4. 3(c), we can see that color property of object only left color, that is black and white. Furthermore, Figure 4. 3(c) does not have problem of overexpose. Same as previous, object hard to see in Figure 4. 3(a).

(a) (b) (c)

Figure 4. 3: Image captured for grayscale return color space(a) Low light intensity (b) Normal light intensity (c) High light intensity

Although performance in handling high light intensity is better, this return color will not consider at this moment since color property of decrease that will limit the improvement of algorithm that may need color property.

Experiment 3: Using RGB as video input format and display as figure.

(Default returned color space in MATLAB)

Since default setting for this webcam is RGB, after delete video object built in previous experiment, a new video input is create and preview directly. No return color space should be set.

```
vid = videoinput('winvideo', 1);
```

```
preview(vid)
```

It is not possible to capture image in dark environment at Figure 4. 4(a).

Figure 4. 4(b) can represent each color of object with details. Furthermore, this return color space did not show problem of overexpose, as in Figure 4. 4(c).

(a) (b) (c)

Figure 4. 4: Image captured for grayscale return color space(a) Low light intensity (b) Normal light intensity (c) High light intensity

From this experiment, it is clear to show that this return color is most suitable for this project among three return color space. It does not lose color property and yet can encounter overexpose problem.

Table 4. 1: Summarize of three experiments conduct previously.

Property

YCbCr

Grayscale

RGB

Able to detect object in low light intensity

No

No

No

Color Returned

Multi color

Black and white

Multi color

Able to encounter overexpose

Partially

No

Yes

From Table 4. 1, we can conclude RGB is the most suitable since from human visual view, grayscale return color space will lose its color characteristic since it will threshold the figure into black and white, we will be unable to further recognize an object exist in frame of view due its unique characteristic such as color. YCbCr can be defined as a way to encode RGB information, thus using RGB will keep original characteristic remain unchanged. Using RGB, we still can develop other usage of it.

Since return color space using is RGB, which is default in toolbox. We can ignore the set return color space in MATLAB coding during import the video input object.

Initially, an object will be created to get input from webcam using following MATLAB command, `obj = videoinput(' winvideo', 1)` where 1 is ID number of camera input. After this MATLAB command is executed, an object named as `obj` will be store in workspace of MATLAB.

In order to let the video input object continuously acquire the data, student has to instruct MATLAB by command as following:

```
triggerconfig(obj, ' manual');
```

```
set(obj, ' Tag', appTitle, ' FramesAcquiredFcnCount', 1, ...
```

```
' TimealrFcn', @locFrameCallback, ' TimerPeriod', 0. 01);
```

#### 4. 2. 2 MATLAB M-file

Initially, we have to associate object (video input object) with figure in GUI of MATLAB, if it is already existed, we will use it or else create a new one.

```
ud = get(obj, ' UserData');
```

```
if ~isempty(ud) && isstruct(ud) && isfield(ud, ' figureHandles') ...
```

```
&& ishandle(ud. figureHandles. hFigure)
```

```
appdata. figureHandles = ud. figureHandles;
```

```
figure(appdata. figureHandles. hFigure)
```

```
else
```

```
appdata. figureHandles = localCreateFigure(obj, appTitle);
```

```
end
```

An empty array with unset dimension and value will be used to store what the video input object needs in terms of application data.

```
appdata. background = [];
```

```
obj. UserData = appdata;
```

Function named as `imaqmotion` which contain MATLAB command will be compile together and compile to ensure no error detect. In order to execute this function, user can create a video input object and executed it by named of function follow by name of video input object in bracket.

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>



#### 4. 2. 3 Error Catching in M-file

To prevent MATLAB contain an existing video input object is running, a stop instruction will be included in M-file.

```
stop(obj)
```

This is to ensure that only one new desire video input object will be use to perform the monitoring process. Besides that, MATLAB will show a warning if frame import from webcam takes too long returning. This warning can be skipped by using:

```
warning off imaq: peekdata: tooManyFramesRequested
```

MATLAB will stop responding and quit improperly if error that unpredicted occur during the process. Thus, we have catch the error and only pop out a warning message to indicate user that error been occur and MATLAB can stop the execution of function gracefully.

```
catch
```

```
error(' MATLAB: imaqmotion: error', ...
```

```
sprintf(' IMAQMOTION is unable to run properly. n%s', lasterr))
```

```
end
```

#### 4. 3 Discussion

In this chapter, student demonstrates how a MATLAB connect with webcam and import real time recording to MATLAB. Follow by preparing an

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

environment where declared video input object will be store in workspace of MATLAB, where this object can be use to start the core of project, subtract object from static background. Steps mention before is to ensure user can executed several step in one simple instruction which is store in MATLAB M-file. In the next chapter, student will show how two consecutive frames being compare and spot which is not belong to previous frame (declare as background of frame) in same location of matrices will be show in MATLAB GUI.

## CHAPTER 5 BACKGROUND SUBTRACTION USING FRAME DIFFERENCE 5. 1

### Overview

To achieve objective of this project, detect object which is moving from the view of vision system, we need develop a monitoring system which able to distinguish moving object and static background. This can be done using writing an algorithm using different language such as C programming, Open CV or MATLAB.

In this chapter, background subtraction using frame difference will be implementing along this project to subtract the background. Background subtraction is a general method where as frame difference is a subset of background subtraction which compare the current frame with previous frame and any pixel not belongs to previous frame is consider as moving object. This method been chosen due to its simple operation and can reduce time require to process those frames import from vision system. Frame use as background will be store as array with constant array which contain information of pixel. This array will use as reference, in another, as a

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

background of image which will be compared with next frame capture by vision system in variable of array. After two frames are being compared by using differencing method, object which consider as moving should be show in a window. Due to simple subtraction method, delay in video processing can be reduced.

Those functions contain above ability will be include in M-file. Those instructions will be include in different function so that it can be executed according to flow of project. These include `localFrameCallback` (a function to update image display by video input object), `localUpdateFig` (function that update GUI window using latest data), `localCreateFigure` (function that create and initialize figure), `localCreateBar` (function that create and initialize bar display).

## 5. 2 Initialize and Creating a Background Image

This section is basically discuss how a background image been created if a new video input object been import and update the background image with next frame after previous comparison been made.

In order to initialize background image, a video input object declared in previous chapter will be import to this function, as well as 'UserData'.

```
appdata = get(vid, 'UserData');
```

```
background = appdata. background;
```

After data from video input object been import, a background image can be create using snapshot operation which instruct MATLAB to store the first frame into memory.

```
if isempty(background),  
  
background = getsnapshot(vid);  
  
end
```

To update the background image for next frame, the next frame will be capture and send latest data to function that require it.

```
localUpdateFig(vid, appdata.figureHandles, frame, background);  
  
appdata.background = frame;  
  
set(vid, 'UserData', appdata);
```

### 5. 3 Frame Differencing

In previous section, a function named as localUpdateFig is containing frame difference method. This method will subtract current frame with background and deliver the result to GUI of window. If no moving object is detected, in other term, result of differencing two frames are zero, GUI of MATLAB will deliver blank screen. By subtract two images, which mean subtract the two pixel array with all the exact location it will get the absolute different between two images. Besides that, a bar will be use an indicator for difference gap between current frames with background image.

```
I = imabsdiff(frame, background);
```

In the first step, an empty background image (background) will be taken as references. This example will be show in Figure 5. 1(a). After that, next frame (frame) which contain moving object will be taken as in Figure 5. 1(b). By subtracting these two images, we can obtain absolutes different between two pictures. We can see the difference of two images in Figure 5. 1(c).

Figure 5. 1: Example of frame differencing  
Empty background (b) A new object (c) Difference of two image

Using a simple command, MATLAB can subtract a background from a frame contain moving object.

#### 5. 4 Flushing Data in Memory

When a background image had been snapshot, it will save in memory of MATLAB and waiting to compare with next frame in form of array. This stack will increase by one after a frame difference operation been perform since current frame will be push into memory. We have to flush frame in memory or else we will unable to get data when that memory is full. Flushing can be performed when previous frame is not required. After flushing operation been done, we can assure that every frame capture by vision system can save to memory and been compared.

```
frame = peekdata(vid, 1);
```

```
if isempty(frame),
```

```
return;
```

```
end
```

```
flushdata(vid);
```

Figure 5. 2: Memory of cache been flush

After memory of cache been flush when those frames are not required, MATLAB can insert new frame coming from vision system continuously as show in Figure 5. 2.

### 5. 5 Terminate Monitoring System Gracefully

MATLAB can either exit with force (without saving data) or exit with gracefully with event handle been set in function. In this part, an event button will be introduced so that user can terminate the monitoring system without harming data which is stored in memory of MATLAB workspace.

Monitoring system can be either exit when user click on close button for GUI window or video input object been deleted in workspace of MATLAB. Error will come out showing that monitoring system do not have video input object if video input object been delete at MATLAB's workspace.

In order to encounter this discourage action toward Monitoring System, this monitoring system should do nothing when video input object been deleted or no longer running, and return this value to function as required. After that, a checking function should be performed when updating figure in GUI. If that entire function receive parameter with value zero (zero at here does not

mean detect nothing but stop responding or terminate by user), it will stop that entire monitoring system. If that function still continuously receives value from parameter, it should continuously perform appropriate action towards input.

## 5. 6 Plotting Result by GUI

GUI is a good tool to let user observe the continuous frames coming from vision system since figure only can show limited figure but GUI can solved this problem. Although GUI have to make many pre-configuration in order to make it consistent, but this GUI can produce better result compare to figure plotting.

A spot for the image object display will create initially and clean up axes. Besides that a motion detection bar will placed below the GUI and it will indicate movement detect from vision system. Figure data will store and display before a new frame coming from vision system will replaced current one. Besides that, a default patch or default line will be used to mark the level of figure. Example of desire GUI is show in the Figure 5. 3.

Title of GUI

Video input frame

Motion Detection Bar

Figure 5. 3: GUI window layout design5. 7 Discussion

In this chapter, core of project that is frame differencing method been implement at video input coming from vision system. From the beginning, how a background image is created (if empty) or been update will use as reference to compare with next frame. This is an important step since without reference, we cannot subtract the background image belong to static frame. Core of this project is subtracting two consecutive frames and any difference between these two frames can consider as moving object.

Since MATLAB will store frame received from vision system continuously to buffer, there is a need to empty the buffer when we does not require using it any more. This can prevent buffer from overflow which can cause data loss for further processing.

In real case, we cannot predict how and when user will terminate the monitoring system, we can consider both cases that either user click on close button for GUI display window or delete the video input object. A termination method will be used for these cases to let this monitoring system terminate with gracefully. This can prevent data loss in MATLAB.

For last part in this chapter, a GUI window is generally discuss and GUI window property being set by student. This window will deliver frame from vision system. A motion detection indication bar also been included in GUI to let user easily found that object in moving is detect.

In next chapter, result obtain from this method will be discussed. Result such as picture details will be included and been discussed. Hence, student can prove this algorithm is fast enough and is most suitable for real time tracking purpose.

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>



## CHAPTER 6 Image Output and Result Obtain

### 6. 1 Overview

In this chapter, it will show the result of moving object detection for this tracking system. Ideal moving object detection and surround factor impact such as wind also will be included.

After webcam is set up, it will synchronize with MATLAB. Some experiments are carrying out to prove it can detect moving object. Until so far, this system only can detect moving object but not tracking moving object.

### 6. 2 Frame without Moving Object

Since this algorithm is subtracting two consecutive frames, and when there is no moving object exist, result obtain is zero because the array of two frames is the same.

In GUI of MATLAB produce, if there is no moving object detected, it will display an empty window with plain black color. The motion detection bar also did not increase to show that there is no moving object exists in this example.

Figure 6. 1: Display figure when there is no moving object

### 6. 3 Frames with Disturbance

Since this detection is depends on frame difference. Hence, it is sensitive to distortion such as moving leaves at tree, small movement cause by wind, moving curtain at shop and etc.

Figure 6. 2: Moving curtain cause by wind

### 6. 4 Ideal Moving Object

If surrounding is a close space and did not have too much disturbance factor, detection of moving object can perfect. Example as follow show a static fan which been switching on and motion of fan been detected. We can see that detection of moving fan is perfectly detected.

Figure 6. 3: Moving stand fan motion. Frame start from up to bottom and left to right. 6. 5 Discussion

In this chapter, result obtain been show and several cases been shown. Problem such as wind causing background of image could not been subtract perfectly. This problem can be solved by improving current algorithm implement. Tracking algorithm also will be included later to make sure it can track object such as vehicle.

In next chapter, conclusion of final year project 1 will be describe and future work that will included for final year project 2 also will be discuss in details.

## CHAPTER 7 CONCLUSION AND FUTURE WORK

### 7. 1 Overview

This chapter will summarize the final year project 1. Achievement, future work and conclusion will be discussed in this chapter as well. In conclusion, vehicle detection been achieve have user can monitor moving object exist from view of vision system. Light changes did not bring too much impact to detection of moving object. Secondly, the software development for tracking system was introduced using detection of moving object. This can prove the whole tracking system able be developed due to detection of moving object already construct.

### 7. 2 Future Work

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

Since achievement of moving object detection been obtain, this tracking system can be furthered improve by modified and enhance it with property such as able to track vehicle more accurately. Tracking system for this project still not so efficient, perhaps better algorithm can be embedded together with algorithm using now to improve ability to track the vehicle which is moving in view.

## References

Al-amri, S. S., Kalyankar, N. V. & Khamitkar, S. D. 2010. A Comparative Study of Removal Noiuse from Remote Sensing Image. Journal of Computer Science Issue. 7(1): 32-36. IJCSI.

Donald, J. S. & James, S. T. 1998 PDMA-2: The Feedback Kalman Filter and Simultaneous Multiple Access of a Single Channel. IEEE Transactions On Circuit and Systems – I: Fundamental Theory and Application. 45(2): 142-149.

Fukushima, H. & Jun-ichi, W. 1991. An Image Processing Technique For background Subtracttion and Its Application to Comet Austin 1989cl. Publication National Astronomy Observatory of Japan. 2: 185-191.

Ganoun, A., Ould-dris, N. & Canals, R. 2006. Tracking System Using Camshift and Feature Points. 14th European Signal Processing Conference. Florence, Italy. EURASIP.

Gurbuz, A. C., McClellan, J. H., Romberg, J. & Scott, W. R. 2008. Compressive Sensing of Parameterized Shapes in Images. 2008 IEEE Internation Conference on Acoustics, Speech and Signal Processing. IEEE. 1949-1952.

<https://assignbuster.com/multiple-objects-tracking-via-collaborative-background-subtraction-computer-science-essay/>

Jia, Y. Q. & Zhang, C. S. 2008. Front-view Vehicle Detection by Markov Chain Monte Carlo Method. Pattern Recognition. 42: 313-321.

Logitech QuickCam Pro 4000 User Manual. 2004.

Neoh, H. S. & Asher, H. 2005 Adaptive Edge Detection for Real-Time Video Processing using FPGAs. Altera.

Nguyen, K., Yeung, G., Ghiasi, S. & Sarrafzadeh, M. 2002. A General Framework for Tracking Objects in a Multi-Camera Environment. International Workshop on Digital and Computational Video. IEEE. 200-204.

Reulke, R., Bauer, S., Doring, T. & Meysel, F. 2007. Traffic Surveillance using Multi-Camera Detection and Multi-Target Tracking. Proceeding of Image and Vision Computing New Zealand. University of Waikato Hamilton, New Zealand. 175-180.