Contents

LIST OF FIGURES

1. INTRODUCTION

1.1 PROBLEM DEFINITION

In today's society, having the ability to detect potential terrorism agents is crucial to safe and effective emergency response. Terrorists or persons intending to wreak havoc have many potential weapons in their arsenals. Fortunately, as the war against terrorism has heated up there have been a number of advances in the field of terrorist agent detection. Increasingly sophisticated and effective equipment is becoming available. As part of their training, emergency responders need to learn how to select and operate a variety of these detection devices.

Without detection devices, emergency responders may have to assume that a terrorist agent is in fact present, and at levels high enough to cause harm. With the advent of detection devices, this is no longer necessary; as technology can be used determine what is really happening. Using the proper detection devices, combined with accurate risk assessment, emergency responders no longer need to be making guesses, and can make informed decisions about personal protective equipment, rescue operations, decontamination, and incident severity.

As complicated and scientific as these devices are, they are not able to interpret the results. It is still up to a human to make decisions.

Hence there arises the need for design of a system which not only detects any suspicious object but also help tracking of person possessing it without any need of any human intervention. The final detected object can not only be displayed but also stored for future reference.

1.2 CONCEPTUAL BLOCK DIAGRAM

The following figure shows the conceptual block diagram of a Suspicious Object Detection and Tracking:-

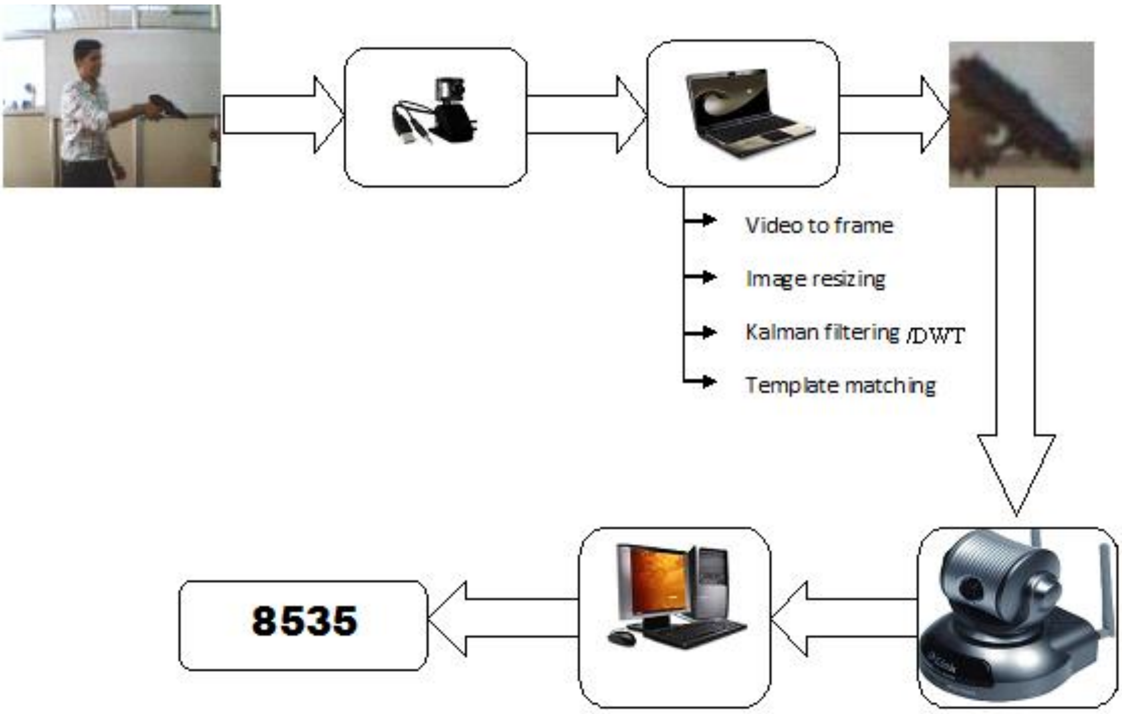


Figure 1 SODT Conceptual Block Diagram

A Suspicious Object Detection and Tracking processing system consists of following:

- Web Camera
- SODT System
- IP Camera
- Personal Computer or a monitoring device
- Atmega 8535 micro-controller
- Application Software (MATLAB)

The real time video signals are acquired from the web camera. These video signals are given as an input to the Laptop where the following functions are performed:-

1. The video acquired from the web camera are converted into frames.
2. These frames are then resized to a fixed size (240,320).
3. Kalman filter/DWT is then applied to these frames in order to detect the variation in the frame position i.e. the motions in the video is detected.
4. Once the motion is detected, template matching program is applied in order to detect the suspicious object. Prior to that, the templates of the various suspicious objects are made and stored in the matlab help.
5. After the detection of the suspicious object the siren is rung. Also the corresponding frames and suspicious object is also stored.

After the detection of the suspicious object, i.e. after the siren is run, the camera placed on the car is turned on and the person carrying the suspicious object is tracked by the manually control provided on the personal computer by means of a Bluetooth control.

1.3 SCOPE OF THE PROJECT

The application provides efficient “Suspicious Object Detection” along with tracking of the suspect from a real time video. For this to happen the algorithm involved includes Template Matching for peculiar object detection and various operations including Kalman Filtering/DWT and background subtraction for Motion detection. However the tracking system includes a Microcontroller System fed by the available (detected) data. The system finds its application at places where real time Video Surveillance is required such as Hotels, Motels and houses etc.

1.3.1 ASSUMPTION

The input video will only be of avi format and the template of the object to be detected will be already present in the database of the user.

1.3.2 LIMITATIONS

The following assumptions are to be made before using the developed system:

1) The background of the video that is given as the input to the system should remain same for most of the time in case of Kalman filtering but it can be overcome by DWT.
2) The size of the template of the object that is to be detected should be less than the actual object.
3) Proper illumination conditions should always be present.
4) There should not be any drastic change in the motion of the suspect to be detected.

2. OBJECT DETECTION TECHNIQUE

2.1 INTRODUCTION

The most common approach to generic object detection/ localization is to slide a window across the image (possibly at multiple scales), and to classify each such local window as containing the target or background. This approach has been successfully used to detect rigid objects such as faces and cars, and has even been applied to articulated objects such as pedestrians.

A natural extension of this approach is to use such sliding window classifiers to detect object parts, and then to assemble the parts into a whole object.

Another popular approach is to extract local interest points from the image, and then to classify each of the regions around these points, rather than looking at all possible sub windows.

Major object detection algorithms, can be roughly classified as the following figure:

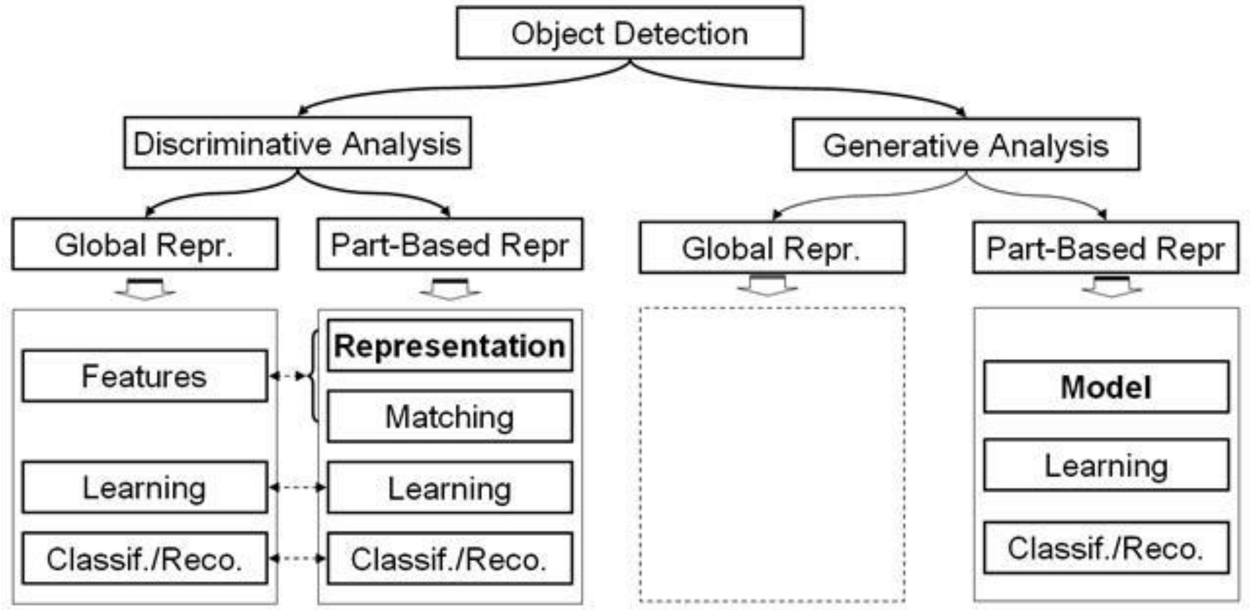


Figure: 2 Overview of Object Detection Methods

In our method of detection of the objects (generally guns like AK47, rifle, bazooka, etc), we are using template matching algorithm which uses normal cross-correlation technique. Correlation is an important tool in image processing, pattern recognition, and other fields. The correlation between two signals (cross correlation) is a standard approach to feature detection as well as a building block for more sophisticated recognition techniques. Textbook presentations of correlation commonly mention the convolution theorem and the attendant possibility of efficiently computing correlation in the frequency domain via the fast Fourier transform.

2.2 KALMAN FILTERING

The **Kalman filter** is a mathematical method named after Rudolf E. Kalman. Its purpose is to use measurements that are observed over time that contain noise (random variations) and other inaccuracies, and produce values that tend to be closer to the true values of the measurements and their associated calculated values. The Kalman filter has many applications in technology, and is an essential part of the development of space and military technology. Perhaps the most commonly used type of very simple Kalman filter is the phase-locked loop, which is now ubiquitous in FM radios and most electronic communications equipment. Extensions and generalizations to the method have also been developed.

The Kalman filter produces estimates of the true values of measurements and their associated calculated values by predicting a value, estimating the uncertainty of the predicted value, and computing a weighted average of the predicted value and the measured value. The most weight is given to the value with the least uncertainty. The estimates produced by the method tend to be closer to the true values than the original measurements because the weighted average has a better estimated uncertainty than either of the values that went into the weighted average.

The filter is named after Rudolf E. Kalman, though Thorvald Nicolai Thiele and Peter Swerling developed a similar algorithm earlier. Stanley F. Schmidt is generally credited with developing the first implementation of a Kalman filter. It was during a visit of Kalman to the NASA Ames Research Center that he saw the applicability of his ideas to the problem of trajectory estimation for the Apollo program, leading to its incorporation in the Apollo navigation computer. This Kalman filter was first described and partially developed in technical papers by Swerling (1958), Kalman (1960) and Kalman and Bucy (1961).

Kalman filters have been vital in the implementation of the navigation systems of U.S. Navy nuclear ballistic missile submarines; and in the guidance and navigation systems of cruise missiles such as the U.S. Navy's Tomahawk missile; the U.S. Air Force's Air Launched Cruise Missile; It is also used in the guidance and navigation systems of the NASA Space Shuttle and the attitude control and navigation systems of the International Space Station.

This digital filter is sometimes called the Stratonovich–Kalman–Bucy filter because it is a special case of a more general, non-linear filter developed somewhat earlier by the

Soviet mathematician Ruslan L. Stratonovich. In fact, some of the equations of the special case linear filter appeared in these papers by Stratonovich that were published before summer 1960, when Kalman met with Stratonovich during a conference in Moscow.

The **Kalman filter** uses a system's dynamics model (i.e. physical laws of motion), known control inputs to that system, and measurements (such as from sensors) to form an estimate of the system's varying quantities (its state) that is better than the estimate obtained by using any one measurement alone. As such, it is a common sensor fusion algorithm.

All measurements and calculations based on models are estimates to some degree. Noisy sensor data, approximations in the equations that describe how a system changes and external factors that are not accounted for introduce some uncertainty about the inferred values for a system's state. The Kalman filter averages a prediction of a system's state with a new measurement using a weighted average. The purpose of the weights is that values with better estimated uncertainty are "trusted" more. The weights are calculated from the covariance, a measure of the estimated uncertainty of the prediction of the system's state. The result of the weighted average is a new state estimate that lies in between the predicted and measured state, and has a better estimated uncertainty than either alone. This process is repeated every time step, with the new estimate and its covariance informing the prediction used in the following iteration. This means that the Kalman filter works recursively and requires only the last "best guess" - not the entire history - of a system's state to calculate a new state.

When performing the actual calculations for the filter (as discussed below), the state estimate and covariance are coded into matrices to handle the multiple dimensions involved in a single set of calculations. This allows for representation of linear relationships between different state variables (such as position, velocity, and acceleration) in any of the transition models or covariance’s.

The **Kalman filter** is an efficient recursive filter that estimates the internal state of a linear dynamic system from a series of noisy measurements. It is used in a wide range of engineering and econometric applications from radar and computer vision to estimation of structural macroeconomic models, and is an important topic in control theory and control systems engineering. Together with the linear-quadratic regulator (LQR), the Kalman filter solves the linear-quadratic-Gaussian control problem (LQG). The Kalman filter, the linear-quadratic regulator and the linear-quadratic-Gaussian controller are solutions to what probably are the most fundamental problems in control theory. In most applications, the internal state is much larger (more degrees of freedom) than the few "observable" parameters which are measured. However, by combining a series of measurements, the Kalman filter can estimate the entire internal state.

In control theory, the Kalman filter is most commonly referred to as **linear quadratic estimation** (LQE).

In Dempster-Shafer theory, each state equation or observation is considered a special case of a linear belief function and the Kalman filter is a special case of combing linear belief functions on a join-tree or Markov tree.

A wide variety of Kalman filters have now been developed, from Kalman's original formulation, now called the *simple* Kalman filter, the Kalman-Bucy filter, Schmidt's *extended* filter, the *information* filter, and a variety of *square-root* filters that were developed by Bierman, Thornton and many others. Perhaps the most commonly used type of very simple Kalman filter is the phase-locked loop, which is now ubiquitous in radios, especially frequency modulation (FM) radios, television sets, satellite communications receivers, outer space communications systems, and nearly any other electronic communications equipment.

2.3 DISCRETE WAVELET TRANSFORM (DWT)

In numerical analysis and functional analysis, a **Discrete Wavelet Transform** (DWT) is any wavelet transform for which the wavelets are discretely sampled. As with other wavelet transforms, a key advantage it has over Fourier transforms is temporal resolution: it captures both frequency and location information (location in time).

An example of the 2D discrete wavelet transform that is used in JPEG2000. The original image is high-pass filtered, yielding the three large images, each describing local changes in brightness (details) in the original image. It is then low-pass filtered and downscaled, yielding an approximation image; this image is high-pass filtered to produce the three smaller detail images, and low-pass filtered to produce the final approximation image in the upper-left.

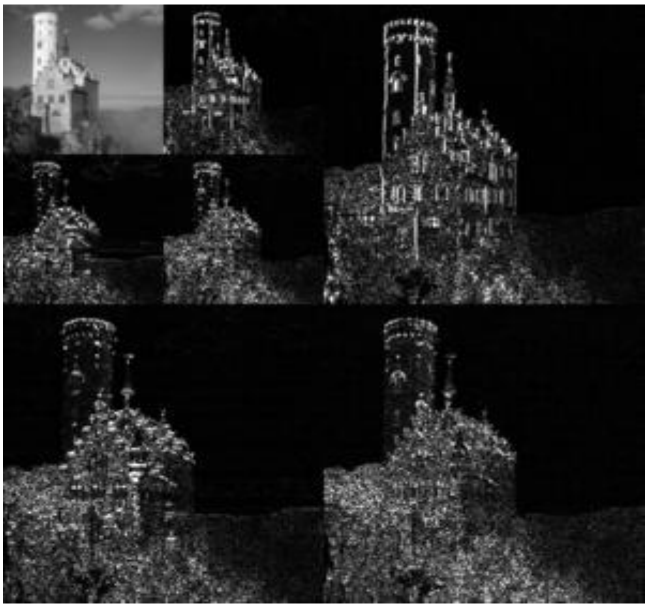


Figure 3 DWT in jpeg 2000

The DWT of a signal *x* is calculated by passing it through a series of filters. First the samples are passed through a low pass filter with impulse response *g* resulting in a convolution of the two:

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k]g[n-k].$$

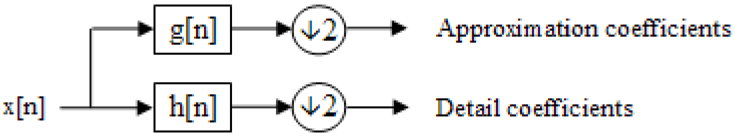
The signal is also decomposed simultaneously using a high-pass filter *h*. The outputs giving the detail coefficients (from the high-pass filter) and approximation coefficients (from the low-pass). It is important that the two filters are related to each other and they are known as a quadrature mirror filter.

However, since half the frequencies of the signal have now been removed, half the samples can be discarded according to Nyquist’s rule. The filter outputs are then subsampled by 2 (It should be noted that Mallat's and the common notation is the opposite, g- high pass and h- low pass):

$$y_{\mathrm{low}}[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n-k]$$

$$y_{\mathrm{high}}[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n+1-k]$$

This decomposition has halved the time resolution since only half of each filter output characterizes the signal. However, each output has half the frequency band of the input so the frequency resolution has been doubled.



Block diagram of filter analysis

With the sub sampling operator $\downarrow$

$$(y \downarrow k)[n] = y[kn]$$

The above summation can be written more concisely.

$$y_{\mathrm{low}} = (x * g) \downarrow 2$$

$$y_{\mathrm{high}} = (x * h) \downarrow 2$$

However computing a complete convolution *x* * *g* with subsequent down sampling would waste computation time.

The Lifting scheme is an optimization where these two computations are interleaved.

2.3.1 Haar wavelet

The first DWT was invented by the Hungarian mathematician Alfréd Haar. For an input represented by a list of 2^n numbers, the Haar wavelet transform may be considered to simply pair up input values, storing the difference and passing the sum. This process is repeated recursively, pairing up the sums to provide the next scale: finally resulting in (2^n-1) differences and one final sum. The Haar DWT illustrates the desirable properties of wavelets in general. First, it can be performed in $O(n)$ operations; second, it captures not only a notion of the frequency content of the input, by examining it at different scales, but also temporal content, i.e. the times at which these frequencies occur. Combined, these two properties make the Fast wavelet transform (FWT) an alternative to the conventional Fast Fourier Transform (FFT).

2.3.2 Daubechies wavelets

The most commonly used set of discrete wavelet transforms was formulated by the Belgian mathematician Ingrid Daubechies in 1988. This formulation is based on the use of recurrence relations to generate progressively finer discrete samplings of an implicit mother wavelet function; each resolution is twice that of the previous scale. In her seminal paper, Daubechies derives a family of wavelets, the first of which is the Haar wavelet. Interest in this field has exploded since then, and many variations of Daubechies' original wavelets were developed.

2.3.3 Applications

The discrete wavelet transform has a huge number of applications in science, engineering, and mathematics and computer science. Most notably, it is used for signal coding, to represent a discrete signal in a more redundant form, often as a preconditioning for data compression.

2.4 TEMPLATE MATCHING

Template matching is used for many applications in image processing. Cross Correlation is the basic statistical approach to image registration. It is used for template matching or pattern recognition. Template can be considered a sub-image from the reference image, and the image can be considered as a sensed image. The objective is to establish the correspondence between the reference image and sensed image. It gives the measure of the degree of similarity between an image and template. This paper describes medical image registration by template matching based on Normalized Cross-Correlation (NCC) using Cauchy-Schwartz inequality. The algorithm for template matching using NCC is implemented in MATLAB. The algorithm does the template matching and uses the Cauchy-Schwartz's inequality to simplify the procedure. The developed algorithm is robust for similarity measure. An experimental result with medical images registration with noise and without noise is shown in the results section.

Template matching is one fundamental task occurring in countless image analysis applications. Template matching is the process of finding the location of a sub image, called a *template*, inside an image. There are number of methods for image registration. Here we have discussed the template matching application for matching a small image which is a part of big image with given big image. Once a number of corresponding templates are found, their centers are used as corresponding control points to determine the registration parameters. Template matching involves comparing a given template with windows of the same size in an image and identifying the window that is most similar to the template.

The basic template matching algorithm consists in calculating at each position of the image under examination a distortion function that measures the degree of similarity between the template and the image. Then, the minimum distortion, or maximum correlation, position is taken to locate the template into the examined image. Typical distortion measures are the Sum of Absolute Differences (SAD) and the Sum of Squared Differences (SSD). However, as far as template matching is concerned, Normalized Cross Correlation (NCC) is often the adopted for similarity measure due to its better robustness.

In Suspicious Object Detection and Tracking system, we are using template matching using normalized cross-correlation (NCC). In this method, suppose for the given image g, we want to match the template f as shown in Fig.1. Simple method for measuring similarity or mismatch measure is by taking the absolute difference between template f and given image g.

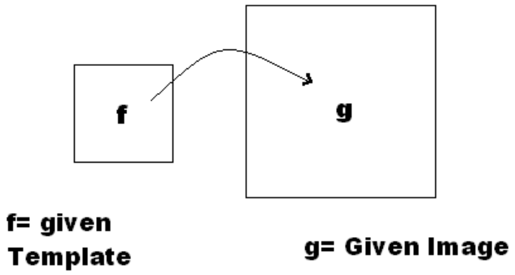


Figure 4 Template Matching f<g

The maximum value of absolute difference will give the similarity measure. Generally *f* is very small image compare to given image *g*. If we take the sum of difference square between template *f* and given image *g* over a region *A*, then in analog form equation is $\iint(f-g)^2$

And in digital form, generally image having 2-D,
$\sum\sum (f(i, j)-g(i, j))^2$
If we expand the above equation (2, 1)
$\iint (f-g)^2 = \iint (f)^2 + \iint (g)^2 - \iint (2fg)$

For given template the term $\iint (f)^2$ is fixed and for given image the term $\iint (g)^2$ is also fixed or constant. The term $\iint (f\text{-}g)^2$ will give the degree of mismatch. If this both terms are constant then remaining term $\iint (2fg)$ is a measure of similarity between template *f* and image *g* [1, 2, 3].

- *Cauchy-Schwartz Inequality*

This inequality is given by

$$\iint (f.\ g) \leq \sqrt{\iint (f)^2 \iint (g)^2}$$

Above equation gives the normalized cross correlation. The maximum value of above ratio will give the measure of similarity in the given image g. The Normalized Cross-covariance is the linear correlation coefficient of statistics and is equivalent to the normalized cross correlation of the zero mean equivalents of *f* and *g* [4, 5]. It is given by

$$C\ (f,\ g) = \frac{n\sum fg - \sum f \sum g}{\sqrt{(n\sum f^2 - g(\sum f)^2)\ (n\sum g^2 - (\sum g)^2)}}$$

Where,

n =number of pixels in the template

f = template

g = image part under template

3. LITERATURE REVIEW

3.1 PRE-PROCESSING

IMAGE RESIZING IN THE DISCRETE COSINE TRANSFORM DOMAIN

Image resizing is a task that must often be done when processing digital images. The paper published by Martucci [2], S. A. consists of novel approach to resize images by operating entirely in the discrete cosine transform (DCT) domain. We implement the low pass filter for anti-aliasing or anti-imaging using the convolution-multiplication property of the DCT. This approach can be used as a standalone image resizing tool or it can be integrated into any image compression system based on a block DCT, such as JPEG, to provide an image resizing capability with relatively little additional complexity

RGB CALIBRATION FOR COLOR IMAGE ANALYSIS IN MACHINE VISION

The paper published by Young-Chang Chang and Reld. J. F. [4] proposed a color calibration method for correcting the variations in RGB color values caused by vision system components was developed and tested in this study. The calibration scheme concentrated on comprehensively estimating and removing the RGB errors without specifying error sources and their effects. The algorithm for color calibration was based upon the use of a standardized color chart and developed as a preprocessing tool for color image analysis. According to the theory of image formation, RGB errors in color images were categorized into multiplicative and additive errors. Multiplicative and additive errors contained various error sources-gray-level shift, a variation in amplification and quantization in camera electronics or frame grabber, the change of color temperature of illumination with time, and related factors. The RGB errors of arbitrary colors in an image were estimated from the RGB errors of standard colors contained in the image. The color calibration method also contained an algorithm for correcting the no uniformity of illumination in the scene. The algorithm was tested under two different conditions-uniform and no uniform illuminations in the scene. The RGB errors of arbitrary colors in test images were almost completely removed after color calibration. The maximum residual error was seven gray levels under uniform illumination and 12 gray levels under no uniform illumination. Most residual RGB errors were caused by residual no uniformity of illumination in images, The test results showed that the developed method was effective in correcting the variations in RGB color values caused by vision system components.

3.2 MOTION DETECTION

AN IMPROVED QUATERNION-BASED KALMAN FILTER FOR REAL-TIME TRACKING OF RIGID BODY ORIENTATION

This paper proposed by Xiaoping Yun, Lizarraga, M., Bachmann, E.R., McGhee [3], R.B., presents an improved Kalman filter for real-time tracking of human body motions. An earlier version of the filter was presented at IROS 2001. Since then, the filter has been substantially improved. Real-time tracking of rigid body orientation is accomplished using the MARG (magnetic, angular rate, and gravity) sensors. A MARG sensor measures the three-dimensional local magnetic field, three-dimensional angular rate, and three-dimensional acceleration. A Kalman filter is designed to process measurements provided by the MARG sensors, and to produce real-time orientation represented in quaternions. There are many design decisions as related to choice of state vectors, output equations, process model, etc. The filter design presented in this paper utilizes the Gauss-Newton method for parameter optimization in conjunction with Kalman filtering. The use of the Gauss-Newton method, particularly the reduced-order implementation introduced in the paper, significantly simplifies the Kalman filter design, and reduces computational requirements.

INTERLACED KALMAN FILTERING OF 3D ANGULAR MOTION BASED ON EULER'S NONLINEAR EQUATIONS

In this approach proposed by Algrain, M. C. and Sanlle, J. , Kalman filtering technique is presented that reduces the mean-square-error (MSE) between three-dimensional (3D) actual angular velocity values and estimated ones by an order of magnitude (when compared with the MSE resulting from direct measurements) even under extremely low signal-to-noise ratio conditions. The filtering problem is nonlinear in nature because the dynamics of 3D angular motion are described by Euler's equations. This nonlinear set of differential equations state that the angular acceleration in one axis is proportional to the torque applied to that axis, and to the products of angular velocity components in the other two axes of rotation. Instead of using extended Kalman filtering techniques to solve this complex problem, the authors developed a new approach where the nonlinear Euler's model is decomposed into two pseudo linear models (primary and secondary). The first model describes the time progression of the state vector containing the linear terms, while the other characterizes the propagation of the state vector containing the nonlinearities. This makes it possible to run two interlaced discrete-linear Kalman filters simultaneously. One filter estimates the values of the state vector containing the linear terms, while the other estimates the values of the state vector containing the nonlinear terms in the system. These estimates are then recombined, solving the nonlinear estimation process without linearizing the system. Thus, the new approach takes advantage of the simplicity, computational efficiency and higher convergence speed of the linear Kalman filter form and it overcomes many of the drawbacks typical of conventional extended Kalman filtering techniques.

MOTION DETECTION USING DWT AND KALMAN FILTER IN MIXED DOMAIN

Katsuia Kondo, Syoji Kobashi and Yutaka Hata proposed a novel method for detection of a moving object in an image sequence. This purpose is achieved by using discrete wavelet transform and extended complex Kalman filter (ECKF) in the transform/spatio-temporal mixed domain. The moving objects are considered as arbitrary trajectory signals (ATS).A trajectory signal in mixed domain is a bunch of 1-dimensional complex signals and an ATS is a bunch of quasi-periodic signals with pitch fluctuations in mixed-domain. A parallel bank of ECKF’s is employed for the estimation of quasi-periodic signals. Experimental results show that the moving object is detected effectively.

3.3 OBJECT DETECTION

FAST AND ROBUST TEMPLATE MATCHING ALGORITHM IN NOISY IMAGE

In the majority of robot applications, including human-computer interaction, template matching is used to find a specific area in a given image or a frame of video stream. Flexible and robust template matching algorithm necessitates feature extraction, for example gradient calculation. This requires complex calculation which causes bad response time of the system. An alternative solution is the use of index table, which stores coordinates that have the same grey level. However, due to the mechanism of the matching algorithm, it is necessary to have several disadvantages in the algorithm. But these restrictions are less important, and there is an idea that copes with these limitations. This paper proposed by Bong Gun Shin, So-Youn Park and Ju Jang Lee, fast and robust template matching algorithm that uses grey level index table and image rank technique. This algorithm can find specific area under the given template query image with 30% Gaussian noise.

OBJECT DETECTION USING HIERARCHICAL MRF AND MAP ESTIMATION

This paper proposed by Qian, R.J. and Huang, T.S. [6] present a new scale, position and orientation invariant approach to object detection. The proposed method first chooses attention regions in an image based on the region detection result on the image. Within the attention regions, the method then detects targets using a novel object detection algorithm that combines template matching methods with feature-based methods via hierarchical MRF and MAP estimation. Hierarchical MRF and MAP estimation provide a flexible framework to incorporate various visual clues. The combination of template matching and feature detection helps to achieve robustness against complex backgrounds and partial occlusions in object detection. Experimental results are given in the paper

4. HARDWARE IMPLEMENTATION

4.1 CIRCUIT DIAGRAM

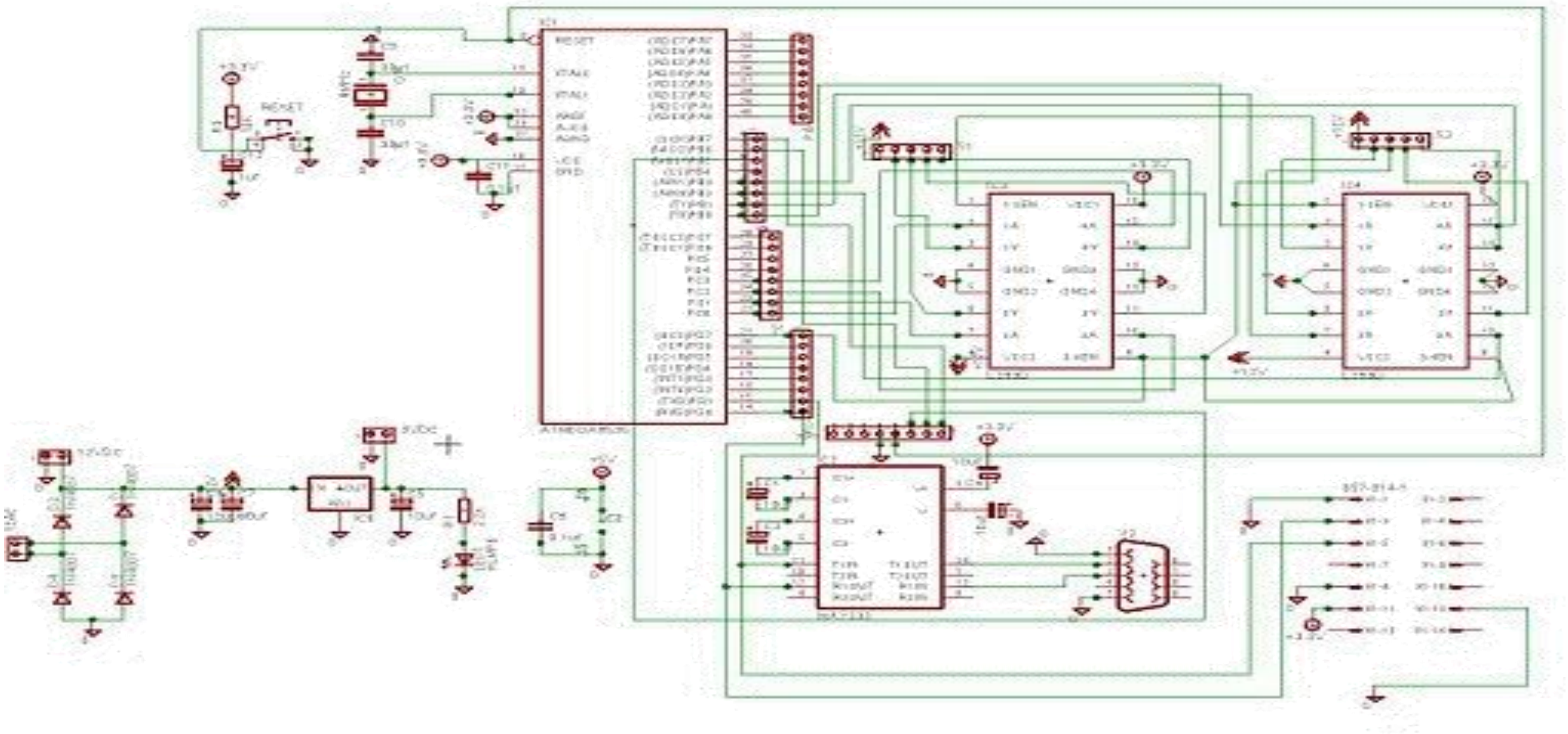


Figure 5: Circuit Diagram

4.2 WORKING

The overall hardware of the project includes a manually controlled robot. The robot will be moved Forward, Backward, Left or Right as per required. The requirement is to track the individual carrying the detected suspicious object. The actual idea is to track the person holding the detected suspicious object unless he/she is caught.

The typical hardware includes the following main modules:-

ATMEGA 8535 MICROCONTROLLER:

The ATmega8535 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing instructions in a single clock cycle, the ATmega8535 achieves throughputs approaching 1 MIPS per MHz allowing the system designed to optimize power consumption versus processing speed.

8535 is a 8-bit microcontroller with following features:

- 130 Powerful Instructions – Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers
- Fully Static Operation
- Up to 16 MIPS Throughput at 16 MHz
- On-chip 2-cycle Multiplier
- It has 4 8bit bi-directional I/O ports with internal pull-up resistors namely port A, B, C and D. The motors are driven by the driver ICs connected to the ports B and C. Port D are used for the Transmission and Reception of the data through RS-232 module.

L239D DRIVER IC

The L293D is a quadruple high-current half-H driver. It is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. It is designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications.

All inputs are TTL compatible. Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN. When an enable input is high, the associated drivers are enabled also their outputs are active and in phase with their inputs. When the enable input is low, those drivers are disabled and their outputs are off and in the high-impedance state. With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

So as per the data given to the microcontroller or say command given by the user to the microcontroller, specific pair of the drivers is activated as per the program fed and the motors are activated thus moving the robot in the desired direction.

H1117S IC (LOW DROPOUT POSITIVE VOLTAGE REGULATOR)

The H1117 Series are available in fixed and adjustable output voltage versions. Over current and thermal overload protection are integrated onto the chip. Output current will decrease while it reaches the preset current or temperature limit. The dropout voltage is specified at 1.2V Maximum at full rated output current. H1117 Series provide excellent regulation over variations due to changes in line, load and temperature. H1117 Series are three terminal regulators and available in popular packages.

The following are the main features of the above mentioned IC:

- Low Dropout Voltage 1.2V at 1.2A
- Adjustable or Fixed Voltage (1.8V, 2.5V, 3.3V, 5V)
- Over Current Protection
- Thermal Overload Protection
- Maximum Line Regulation 0.45%
- Maximum Load Regulation 0.4%
- Adjust Pin Current Less Than 90 uA

The following are the main applications of the IC:

- SCSI-2 Active Termination
- High Efficiency Linear Regulators
- 5V to 3.3V Voltage Converter
- Battery Charger
- Battery Management Circuits for Notebook and Palmtop PCs
- Core Voltage Supply: FPGA, PLD, DSP, CPU

The above IC is included in the circuitry so as to take care that the voltage does not exceed 3.2 to 3.5 volts throughout. Hence here it is used as a linear regulator.

ESD 100 BLUETOOTH MODULE

The transmission of command given to the Microcontroller placed on the robot via the system (PC) is done through the Dongle whereas the command is simultaneously received by the robot by this Bluetooth module placed on it. The camera placed on the robot is a wireless AV camera through which the system receives constant pictures of the person holding the suspicious object. Thus the entire SODT system is made wireless.

4.3 COMPONENT LIST

Component	No. required
Camera 5300G	1
Web camera	1
12 Vdc/ 100 rpm servo motor	4
Atmega 8535 Microcontroller	1
ESD 100 Bluetooth device	1
Motor Driver IC L293D	2
MAX 232	1
H1117S 3.3 Regulator IC	1
Crystal 8MHZ	1
Push button switch	1
D9 Serial connector	1
Diodes (1N4007)	4
LED	1
Resistor 10KΩ	1
2.2KΩ	1
Capacitor 1000µF	1
10µF	5
1µF	1
33pF	2
0.1µF	2

Figure 6: component list

5. SOFTWARE IMPLEMENTATION

5.1 OVERVIEW OF MATLAB

MATLAB stands for "Matrix Laboratory" and is a numerical computing environment and fourth-generation programming language. Developed by The Math Works, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, and FORTRAN.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing capabilities. An additional package, Simulink, adds graphical multi-domain simulation and Model-Based Design for dynamic and embedded systems.

In 2004, Math Works claimed that MATLAB was used by more than one million people across industry and the academic world. MATLAB users come from various backgrounds of engineering, science, and economics.

MATLAB was created in the late 1970s by Cleve Moler, then chairman of the computer science department at the University of New Mexico. He designed it to give his students' access to LINPACK and EISPACK without having to learn FORTRAN. It soon spread to other universities and found a strong audience within the applied mathematics community. Jack little, an engineer, was exposed to it during a visit Moler made to Stanford University in 1983. Recognizing its commercial potential, he joined with Moler and Steve Bangert. They rewrote MATLAB in C and founded The Math Works in 1984 to continue its development. These rewritten libraries were known as JACKPAC. In 2000, MATLAB was rewritten to use a newer set of libraries for matrix manipulation, LAPACK.

MATLAB was first adopted by control design engineers, Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of linear algebra and numerical analysis, and is popular amongst scientists involved with image processing.

MATLAB, the application, is built around the MATLAB language. The simplest way to execute MATLAB code is to type it in at the prompt, >>, in the Command Window, one of the elements of the MATLAB Desktop. In this way, MATLAB can be used as an interactive mathematical shell. Sequences of commands can be saved in a text file, typically using the MATLAB Editor, as a script or encapsulated into a function, extending the commands available.

5.2 ALGORITHM

- **DWT (Discrete Wavelet Transform)**

The algorithm for Discrete Wavelet Transform is implemented in MATLAB. The following algorithm does the Discrete Wavelet Transform:-

1. DWT is applied on background image.
2. Only high frequency components addition and contrast adjustment is applied on the image.
3. Repeat procedure 1 & 2 for all images i.e. frames.
4. Subtract the background image from frame.
5. Apply region filling on this resultant image.
6. Label the different connecting regions.
7. Determine connecting regions parameter.
8. Identification the motion from connecting regions using morphological threshold.
9. Repeat the steps for all frames.

- **Template Matching**

The algorithm for template matching using NCC is implemented in MATLAB. The following algorithm does the template matching and uses the Cauchy-Sehwartz's inequality to simplify the procedure.

1. Load the original image and template.
2. Pad the image on all the sides with zeros so that the centre of the template falls on the very first pixel of the main image when kept on the top-left corner as shown in the figure

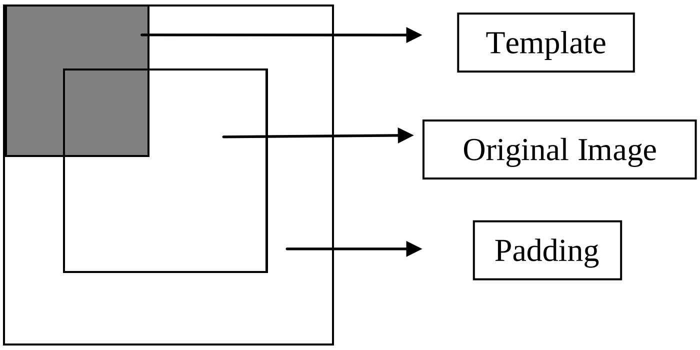


Figure 7 Template Matching

a. Calculate the size of the template.
b. Pad rows of zeros on the top and bottom of main image. The number of rows is equal to the size of template in y-direction divided by 2.
c. Pad columns of zeros on the left and right side of the above image with number of columns equal to the size of the template in z-direction divided by 2 and the length of columns being main image size in y-direction plus size of the template in y-direction.

3. Now, more the mask over the entire image and simultaneously calculate the value of summation of template padded image under the template and store it in an array.
4. Also calculate the values padded image under the template's square and sum all the values. Take the square root of the obtained value and store it in an array.
5. Divide the result obtained in step 3 by the result obtained in step 4.
6. Find the position where the maximum value in the above result falls. The co-ordinates so obtained will give the best match of the template and calculate maximum cross correlation coefficient.
7. Recover the template from the main image using the above obtained co-ordinates and the size of template.

5.3 FLOWGRAPH

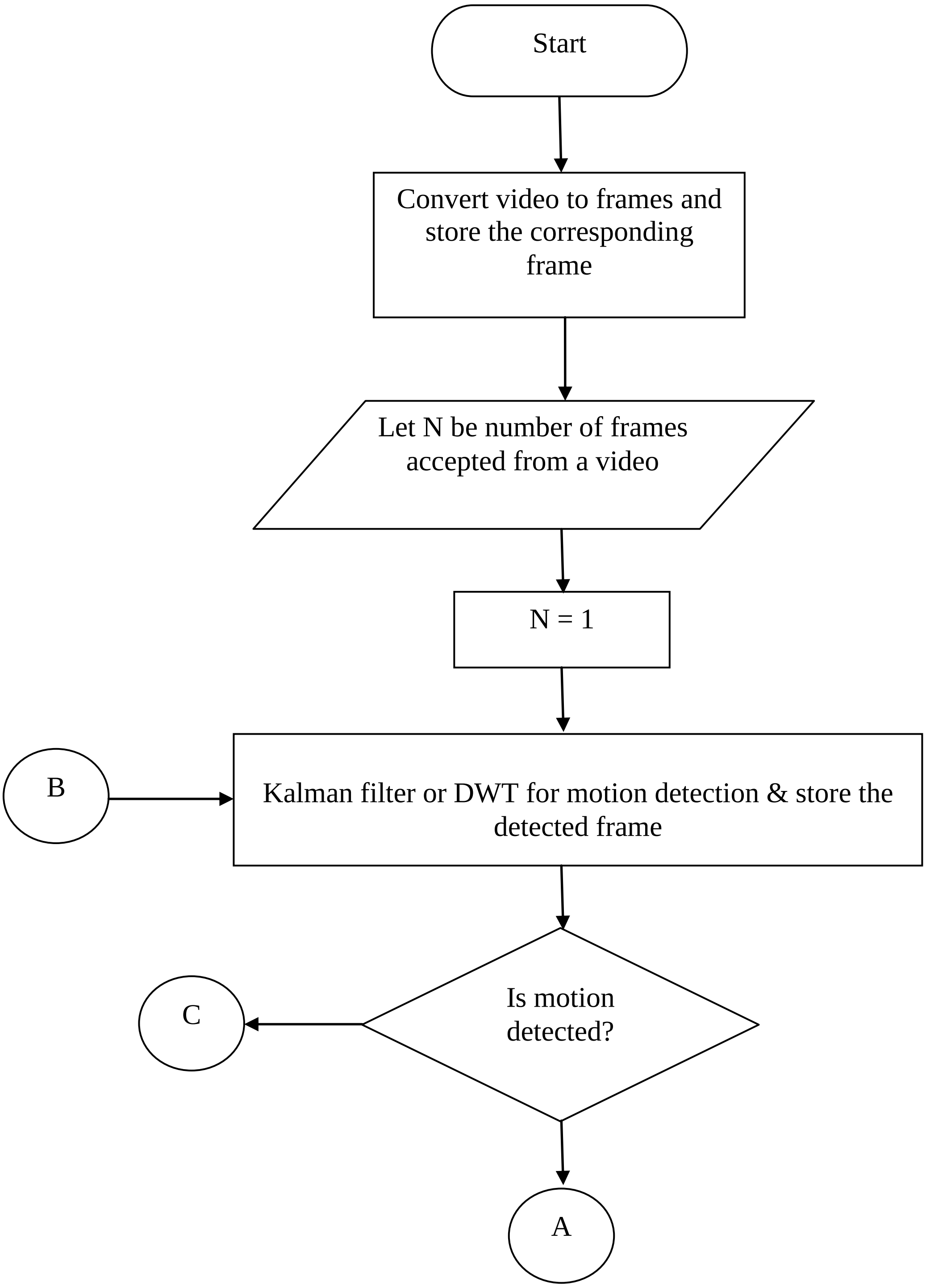

Start
Convert video to frames and store the corresponding frame
Let N be number of frames accepted from a video
N = 1
B
Kalman filter or DWT for motion detection & store the detected frame
C
Is motion detected?
A

Figure 8 Algorithm (a)

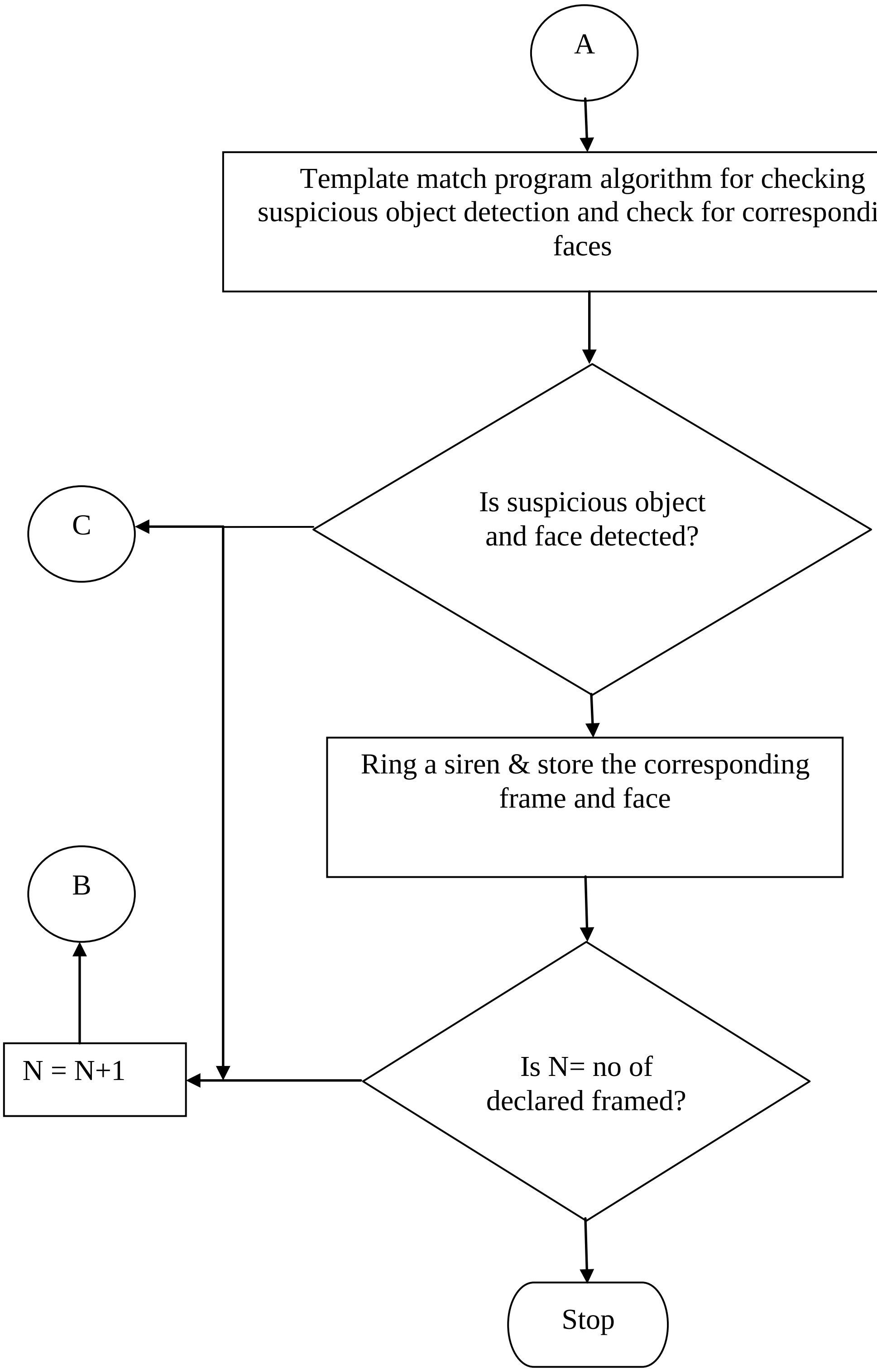

A
Template match program algorithm for checking suspicious object detection and check for corresponding faces
Is suspicious object and face detected?
C
Ring a siren & store the corresponding frame and face
B
N = N+1
Is N= no of declared framed?
Stop

Figure 9 Algorithm (b)

5.4 OVERVIEW OF AVR

- **Creating a New Project**

You can create a new Project using the File |New menu command or by pressing the Create new file button on the toolbar.

A dialog box appears, in which you must select File Type |Project and press the OK button.

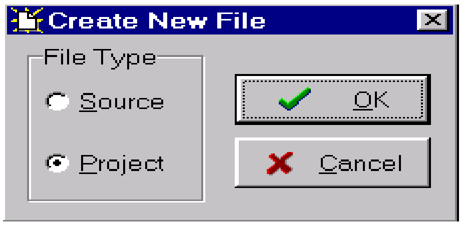


Figure 10 Creating a new project in AVR

The Project file will have the .prj extension.

You can configure the Project by using the Project Configure menu command.

- **Delay Functions**

These functions are intended for generating delays in C programs.

The prototypes for these functions are placed in the file delay, located in the.\INC subdirectory. This file must be #included before using the functions.

Before calling the functions the interrupts must be disabled, otherwise the delays will be much longer than expected. Also it is very important to specify the correct AVR chip clock frequency in the Project|Configure|C Compiler Code Generation menu.

The functions are:

void delay_us (unsigned int n)

generates a delay of n seconds. n must be a constant expression.

void delay_ms (unsigned int n)

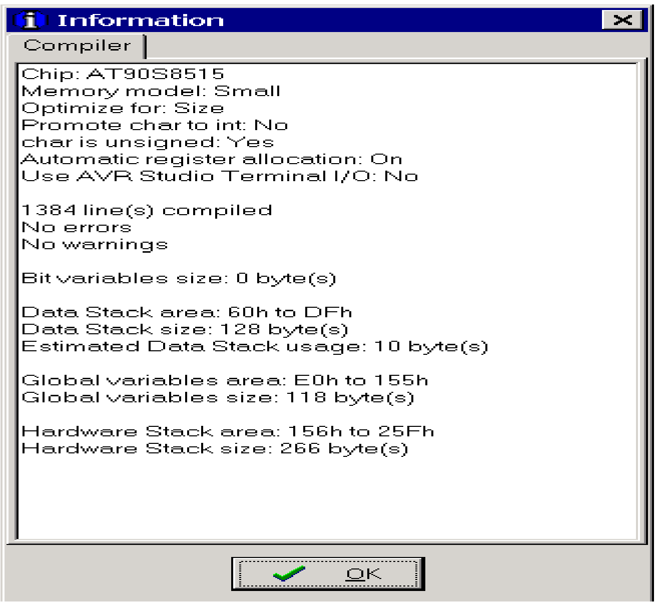
generates a delay of n milliseconds.

- **Compiling the Project**

Assembler source file with the .asm extension. This file can be examined and modified by opening it with the Editor.

The compilation process can be stopped using the Project Stop Compilation menu command or by pressing the Stop Compilation button on the toolbar.

After the compilation is complete, an Information window will open showing the compilation results.



```
Chip: AT90S8515
Memory model: Small
Optimize for: Size
Promote char to int: No
char is unsigned: Yes
Automatic register allocation: On
Use AVR Studio Terminal I/O: No

1384 line(s) compiled
No errors
No warnings

Bit variables size: 0 byte(s)

Data Stack area: 60h to DFh
Data Stack size: 128 byte(s)
Estimated Data Stack usage: 10 byte(s)

Global variables area: E0h to 155h
Global variables size: 118 byte(s)

Hardware Stack area: 156h to 25Fh
Hardware Stack size: 266 byte(s)
```

Figure 11 Compiling the Project

- **Making the Project**

To make the main Project file you must use the Project Make menu command, press the Shift+F9 keys or press the Make button of the toolbar. The CodeVisionAVR C Compiler will be executed, producing an assembler source file with the .asm extension.

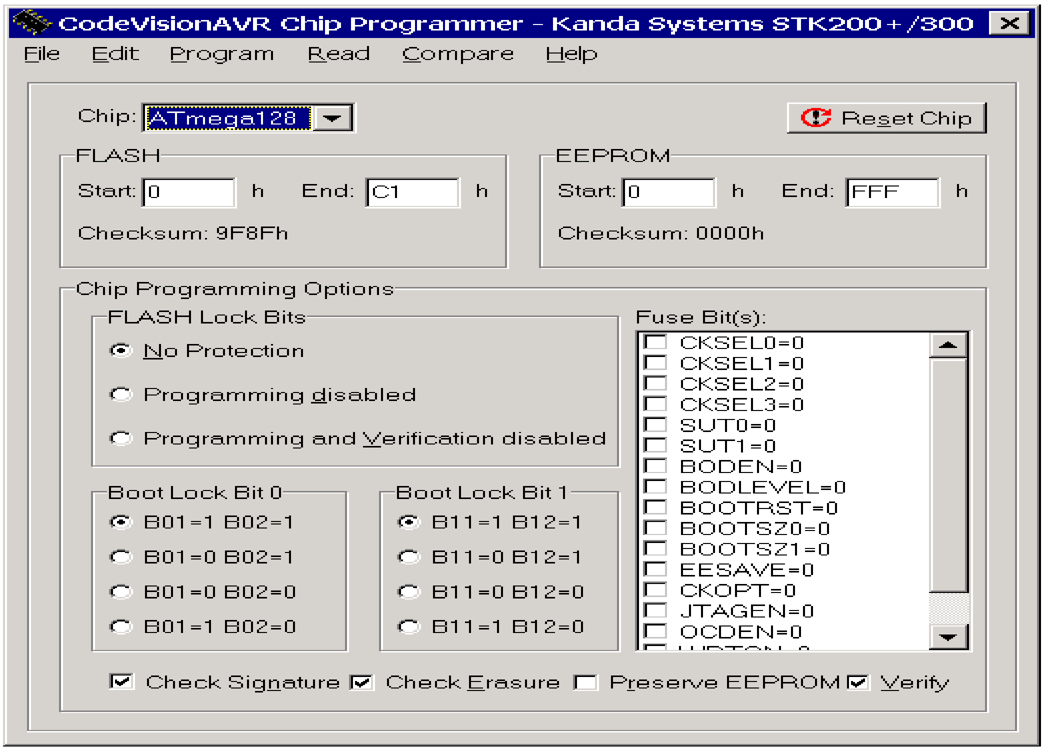
The compilation process can be stopped using the Project Stop Compilation menu command or by pressing the Stop Compilation button on the toolbar. Eventual compilation errors and/or warnings will be listed in the Message window located under the Editor window, or in the Navigator window.

By double clicking on the error or warning message, the line with the problem will be highlighted. If no errors were encountered, then the Atmel AVR assembler AVRASM32 will be executed, obtaining the output file type specified in Project|Configure|C Compiler. After the make process is completed, an Information window will open showing the compilation results. Pressing the Compiler tab will display compilation results.

- **The AVR Chip Programmer**

The CodeVisionAVR IDE has a built-in In-System AVR Chip Programmer that lets you easily transfer your compiled program to the microcontroller for testing. The Programmer is designed to work with the Atmel STK500/AVRISP/AVRProg (AVR910 application note), Kanda Systems STK200+/300, Dontronics DT006, Vogel Elektronik VTEC-ISP, Futurlec JRAVR or the MicroTronics ATCPU/Mega2000 development boards.

The type of the used programmer and the printer port can be selected by using the Settings Programmer menu command. The AVR Chip Programmer is executed by selecting the Tools Chip Programmer menu command or by pressing the Chip Programmer button on the toolbar.


CodeVisionAVR Chip Programmer - Kanda Systems STK200+/300
File Edit Program Read Compare Help
Reset Chip

Chip: ATmega128

FLASH
Start: 0 h End: C1 h
Checksum: 9F8Fh

EEPROM
Start: 0 h End: FFF h
Checksum: 0000h

Chip Programming Options

FLASH Lock Bits
● No Protection
○ Programming disabled
○ Programming and Verification disabled

Boot Lock Bit 0
● B01=1 B02=1
○ B01=0 B02=1
○ B01=0 B02=0
○ B01=1 B02=0

Boot Lock Bit 1
● B11=1 B12=1
○ B11=0 B12=1
○ B11=0 B12=0
○ B11=1 B12=0

Fuse Bit(s):
□ CKSEL0=0
□ CKSEL1=0
□ CKSEL2=0
□ CKSEL3=0
□ SUT0=0
□ SUT1=0
□ BODEN=0
□ BODLEVEL=0
□ BOOTRST=0
□ BOOTSZ0=0
□ BOOTSZ1=0
□ EESAVE=0
□ CKOPT=0
□ JTAGEN=0
□ OCDEN=0

☑ Check Signature ☑ Check Erasure □ Preserve EEPROM ☑ Verify

Figure 12 The AVR Chip Programmer

You can select the type of the chip you wish to program using the Chip combo box.

If the chip you have selected has Fuse Bit(s) that may be programmed, then a supplementary Fuse Bit(s) check box will appear. Using this check box you can set various chip options, which are described in the Atmel data sheets.

If a Fuse Bit(s) check box is checked, then the corresponding fuse bit will be set to 0, the fuse being considered as programmed (as per the convention from the Atmel data sheets).

If a Fuse Bits(s) check box is not checked, then the corresponding fuse bit will be set to 1, the fuse being considered as not programmed.

If you wish to protect your program from copying, you must select the corresponding option using the FLASH Lock Bits radio box.

The Programmer has two memory buffers:

1. The FLASH memory buffer
2. The EEPROM memory buffer.

You can Load or Save the contents of these buffers using the File menu.

Supported file formats are:

1. Atmel .rom and .eep
2. Intel HEX
3. Binary .bin

The Program All menu command allows to automatically:

1. Erase the chip
2. FLASH and EEPROM blank check
3. Program and verify the FLASH
4. Program and verify the EEPROM
5. Program the Fuse and Lock Bits.

- **The Serial Communication Terminal**

The Terminal is intended for debugging embedded systems, which employ serial communication (RS232, RS422, and RS485). The Terminal is invoked using the Tools Terminal menu command or the Terminal button on the toolbar.

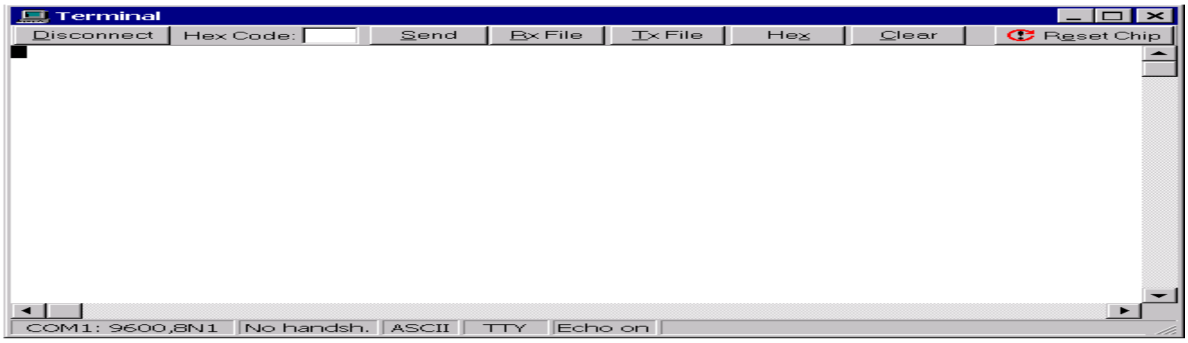


Figure 13 The Serial Communication Terminal

The Disconnect/Connect button allows stop/resume the serial communication.

The characters can be displayed in ASCII or hexadecimal format. The display mode can be toggled using the Hex/ASCII button. The received characters can be saved to a file using the Rx File button. Any characters typed in the Terminal window will be transmitted through the PC serial port. The entered characters can be deleted using the Backspace key.

By pressing the Send button, the Terminal will transmit a character whose hexadecimal ASCII code value is specified in the Hex Code edit box. By pressing the Tx File button, the contents of a file can be transmitted through the serial port. Pressing the Clear button erases the contents of the Terminal window.

5.5 EAGLE SOFTWARE

EAGLE (Easily Applicable Graphical Layout Editor) is a proprietary ECAD program produced by Cadsoft in Germany (American marketing division: Cadsoft USA). It is very commonly used by private electronics enthusiasts, because there is a very usable free demo version for nonprofit use and is available in English and German. Cadsoft has released versions for Microsoft Windows, Linux, and Mac OS X.

- **Features**

EAGLE provides a schematic editor, for designing circuit diagrams and a tightly integrated PCB layout editor, which automatically starts off with all of the components required by the schematic. Components are manually arranged on the board, with the help of colored lines showing the eventual connections between pins that are required by the schematic, to aid in finding a placement that will allow the most efficient track layout.

It also provides a good auto router, which once the components have been placed will attempt to automatically find an optimal track layout to make the electrical connections. It does not always manage to find a way of routing all the signals, although it permits manual routing of critical paths such as power and high frequency lines before letting the auto router handle the other connections. The *.brd* files that EAGLE uses to store board layouts are accepted by many PCB production houses. The EAGLE Layout Editor is an easy to use, yet powerful tool for designing printed circuit boards (PCBs).

The program consists of three main modules

- Layout Editor
- Schematic Editor
- Auto router

which are embedded in a single user interface. Therefore there is no need for converting net lists between schematics and layouts.

Layout Editor

- Maximum drawing area 1.6 x 1.6m (64 x 64 inch)
- Resolution 1/10,000mm (0.1 micron)
- Up to 16 signal layers
- Conventional and smd parts

- Comes with a full set of part libraries
- Easily create your own parts with the fully integrated library editor
- Undo/redo function for any editing command, to any depth
- Script files for batch command execution
- Copper pouring
- Cut and paste function for copying entire sections of a drawing
- Design rule check

Schematic Editor

- Up to 999 sheets in one schematic
- Electrical rule check
- Gate and pin swap
- Create a board from a schematic with a single command

Auto router

- Rip up & retry router
- Up to 16 signal layers
- Routing strategy driven by user definable cost factors

- **Creating the board**

Creating the board from a schematic is one of the easiest tasks with EAGLE. The **Board** command creates a new window with all the parts arranged next to a default board outline. All the nets from the schematic are shown as air wires.

- **Placing the parts**

The **Move** command allows you to pick up each part and move it to its desired position. You can also move entire groups of parts. The right mouse button rotates the current part or group. The **Ratsnest** command recalculates the air wires.

- **Routing the signals**

Annual routing is done with the **Route** command. You simply pick up an air wire, adjust the angle with the right mouse button, and select the layer with the center button. Wires will be added automatically as needed. You can also let the **Auto router** do the routing.

- **Editing the board**

At any time you can

- **Move**, **Rotate** and **Mirror** objects
- **Change** parameters
- **Split** wires etc.

And you never have to worry about messing up things, because you have the most powerful **Undo/Redo** function at hand, which undoes 100% of **every** editing command (unlike other programs, which limit you to *the last delete command* or so).

- **Auto router**

EAGLE comes with an optional rip up & retries Auto router, which is user configurable through cost factors. It can route conventional as well as SMD layouts on up to 16 layers and is fully integrated into the layout editor.

6. RESULT ANALYSIS AND CONCLUSION

6.1 RESULT ANALYSIS

The experiment of Suspicious Object Detection and Tracking was performed on 50 to 100 frames. For the result analysis, we are considering only 50 frames. In these frames, the frames encountering the motion and suspicious object are detected and plotted considering the variation in the templates from 1 to 22.

6.1.1 TEMPLATES

These are some of the templates that are made and saved in the database for the detection of this type of gun from images.

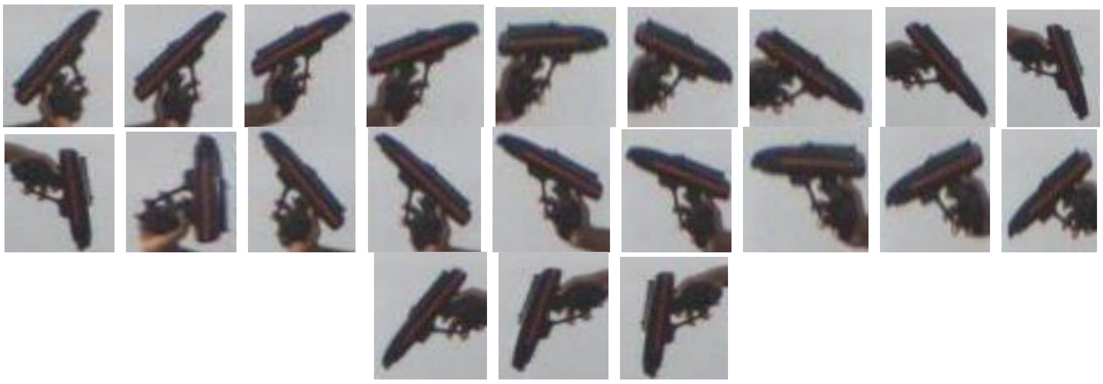


Figure 14 Templates

6.1.2 INPUT FRAMES

The input video is converted into N number of frames as per user requirement and saved into a separate folder for further processing.

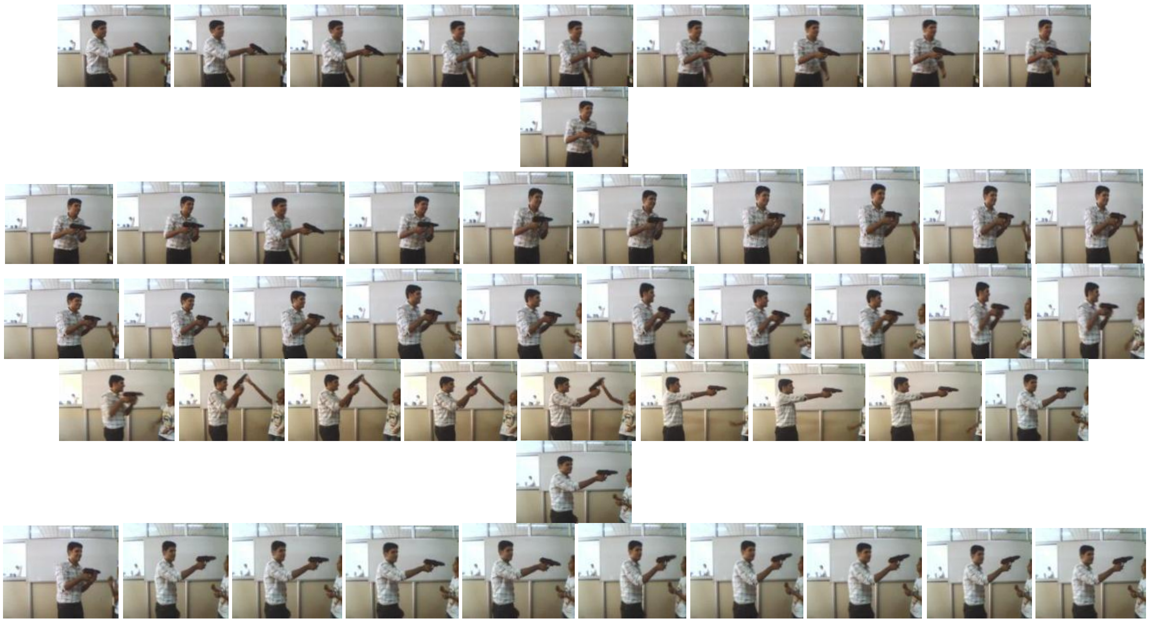


Figure 15 Input Frames

6.1.3 MOTION DETECTED FRAMES

These are the frames where motion is detected into the input frames taken from the video and are also stored separately.

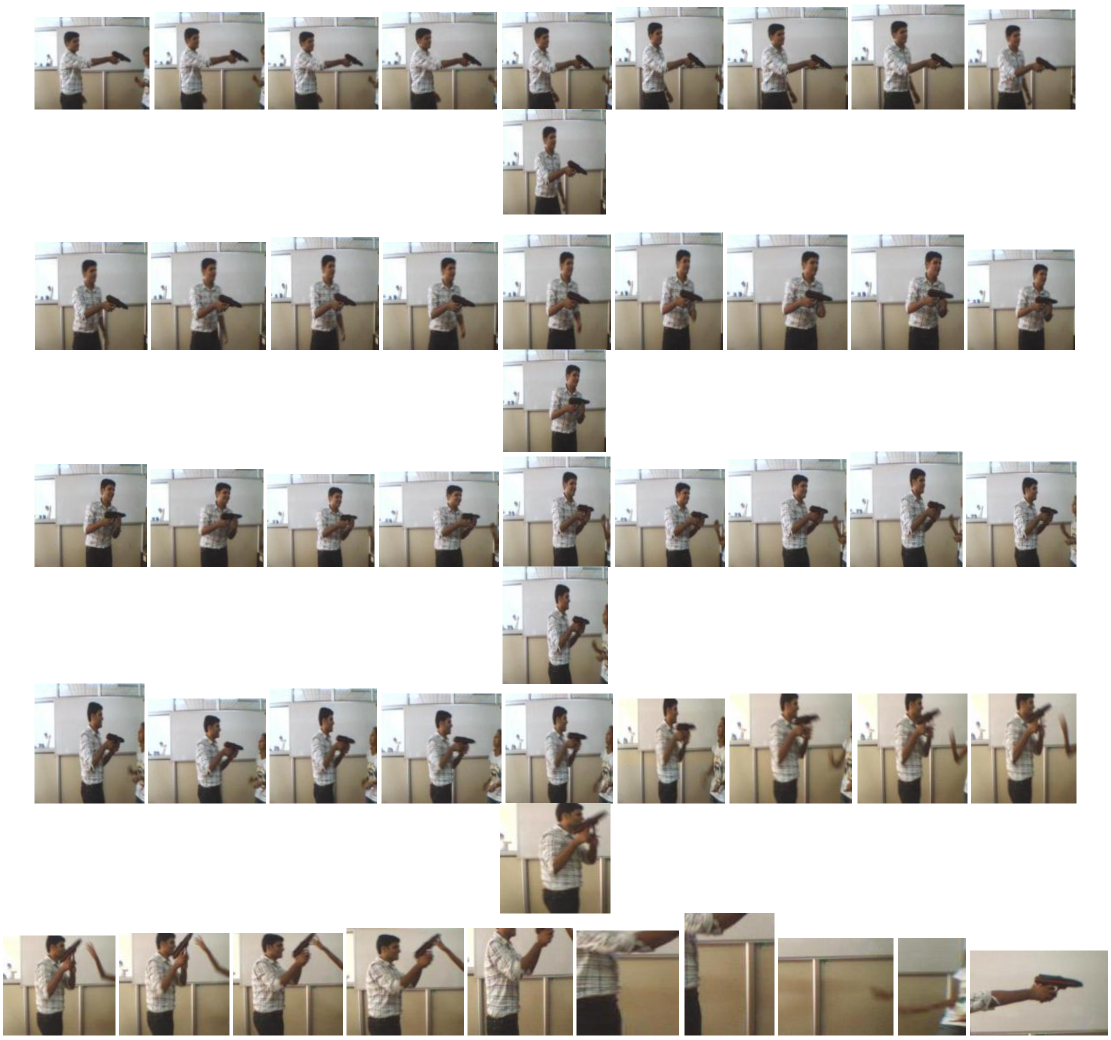


Figure 16 Motion Detected Frames

6.1.4 CROPPED SUSPICIOUS OBJECT

Once the gun is detected using Template Matching Algorithm from the input frames, the detected object (in this case a gun) is cropped and saved into a separate folder.

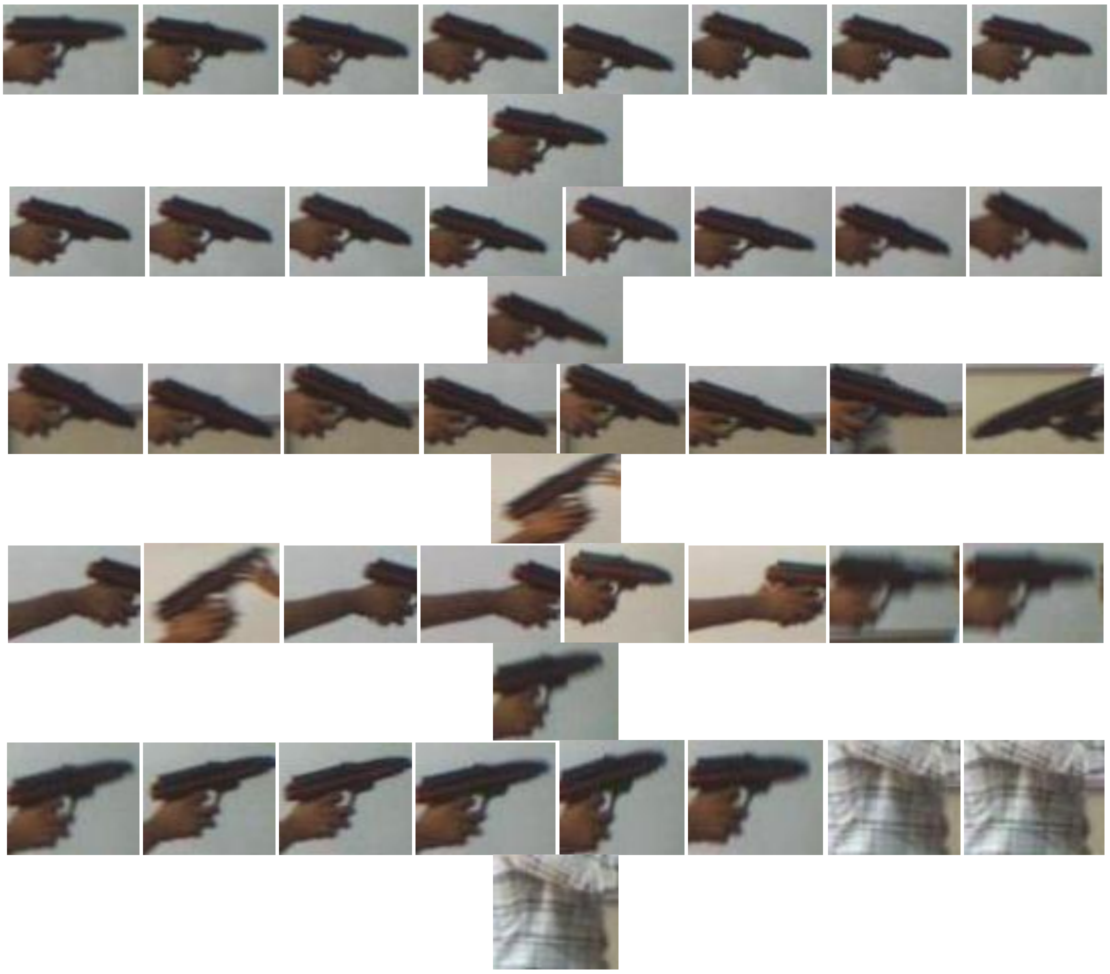


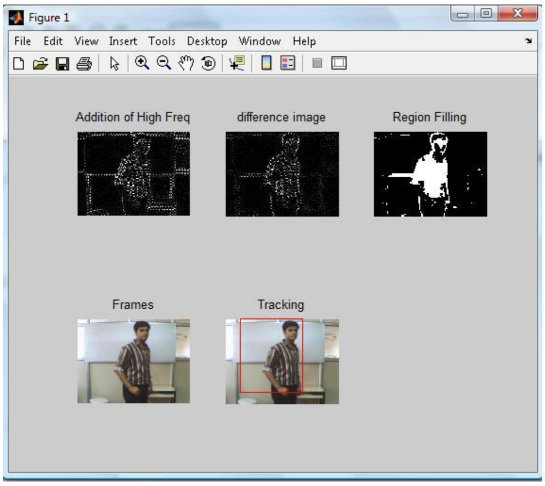
Figure 17 Cropped suspicious object

6.1.5 EXPERIMENTAL DETAILS

The experiment of SODT system is performed on one to many persons, with various combinations of gun i.e. with and without gun and the various conclusions are drawn from them as follows:

CASE 1: ONE PERSON WITHOUT GUN

This is the case wherein a person without a suspicious object (gun) is taken into consideration and the result achieved is that no gun is detected and his motion is however detected.



NO. OF DETECTED GUNS IN FRAME: 0/20

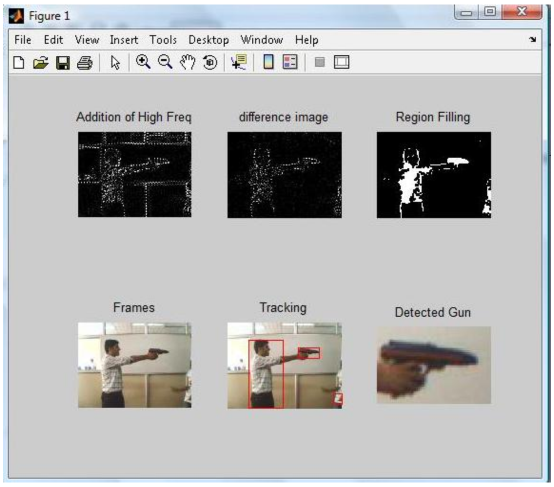
NO. OF FALSE DETECTION: 0/20

NO. OF FRAMES IN WHICH NO DETECTED GUNS: 20/20

Figure 18 Final result (a)

CASE 2: ONE PERSON WITH GUN

A single person with a gun is considered and the result is as shown that the gun and motion both are detected simultaneously.



NO. OF DETECTED GUNS IN FRAME: 10/20

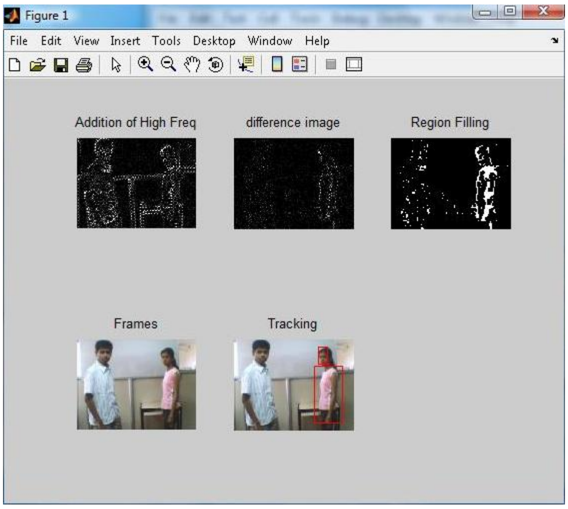
NO. OF FALSE DETECTION: 0/20

NO. OF FRAMES IN WHICH NO DETECTED GUNS: 10/20

Figure 19 Final result (b)

CASE 3: TWO PERSONS WITHOUT GUN

In this case no gun is detected as desired while the motion of both the persons into the frame is detected.



NO. OF DETECTED GUNS IN FRAME: 0/20

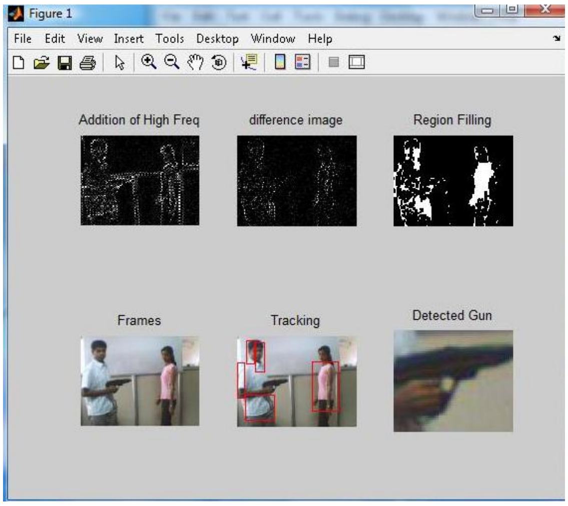
NO. OF FALSE DETECTION: 0/20

NO. OF FRAMES IN WHICH NO DETECTED GUNS: 20/20

Figure 20 Final result (c)

CASE 4: TWO PERSONS, ONE WITH GUN AND ONE WITHOUT GUN

In this case a gun is properly detected and the motion of both the persons in the frame is also detected.



NO. OF DETECTED GUNS IN FRAME: 9/20

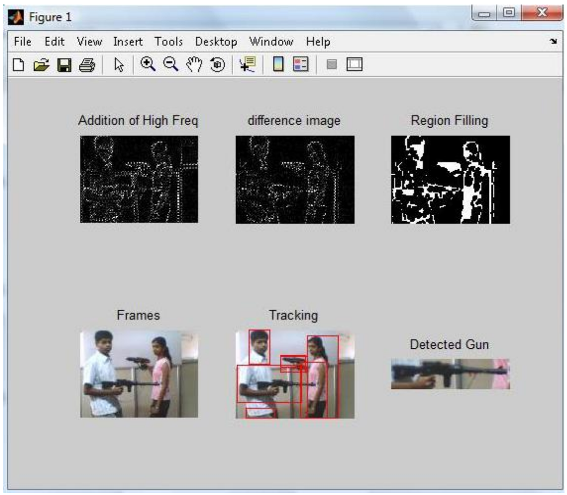
NO. OF FALSE DETECTION: 0/20

NO. OF FRAMES IN WHICH NO DETECTED GUNS: 11/20

Figure 21 Final result (d)

CASE 5: TWO PERSONS, BOTH WITH GUN

Here both the guns as well as motion of both the person are detected. However templates for both the guns should be present in the database.



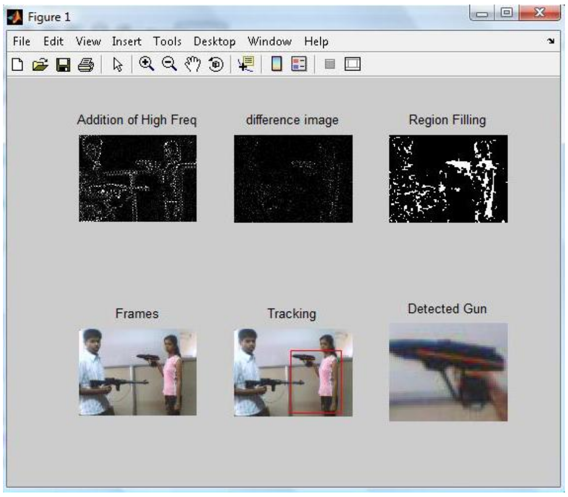


Figure 22 Final result (e)

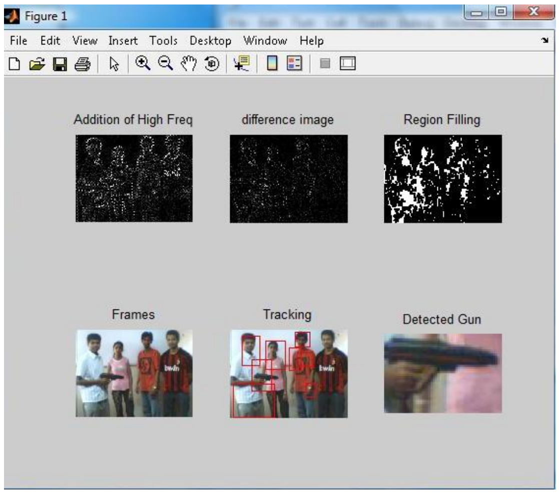
NO. OF DETECTED GUNS IN FRAME: 19/20

NO. OF FALSE DETECTION: 1/20

NO. OF FRAMES IN WHICH NO DETECTED GUNS: 0/20

CASE 6 ONE PERSON HOLDING GUN IN THE CROWD

Here the gun is detected even from the crowd thus giving desired results.



NO. OF DETECTED GUNS IN FRAME: 17/20

NO. OF FALSE DETECTION: 2/20

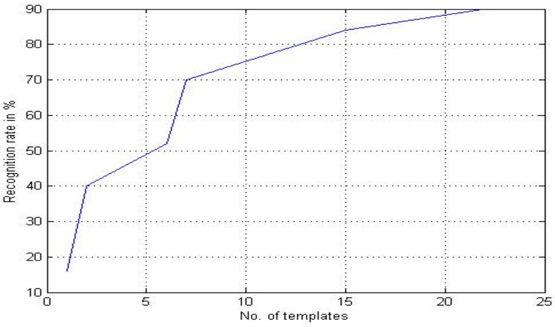
NO. OF FRAMES IN WHICH NO DETECTED GUNS: 1/20

Figure 23 Final result (f)

6.2 GRAPHS

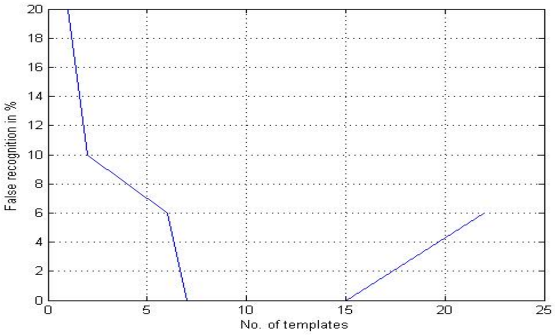
From the graphs displayed below, it is observed that as the no. of templates increases, the recognition efficiency increases with the decrease in not recognized rate. The factor of false detection is invariant of the above two parameters as it depends on the value of normal cross correlation coefficient.

No. of Templates	Recognition rate in %
1	16
2	40
6	52
7	70
15	84
22	90



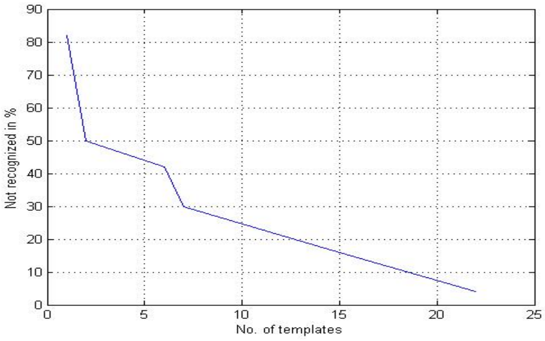
(a) Recognition Efficiency

No. of Templates	False Recognition in %
1	20
2	10
6	06
7	00
15	00
22	06



(b) Failure Recognition Rate

No. of Templates	Not Recognized in %
1	82
2	50
6	42
7	30
15	16
22	04



(c) Not Recognized Rate

Figure 24: Graph of different parameters of SODT system

7. CONCLUSION

The proposed system is used to detect the suspicious object and motion of the person holding it using various algorithms. This system is invariant to lighting condition, background scenario and view point. Once the object (gun in our case) is detected using this system, and if the person holding it goes out of range of the still camera then a hardware system (robot) is designed to follow the respective person and thus tracking of the suspected person is achieved.

The algorithms used in the designed SODT system are DWT (Discrete Wavelet Transform) for motion detection and Template Matching using Cross Correlation method for object detection. However the main disadvantage of using Template Matching Algorithm is that a large database is required where the templates of the gun to be detected need to be stored else the object (gun) will not be detected. Hence if you want to detect all the guns, then the templates of each type of gun should be present. Also, more the number of templates, more is the time required in matching and thus giving the final results.

In case of Kalman filtering, as the background is varied, we get undesired results. This problem is avoided by using DWT. However the results can be further improved using "Weiner Filter", where sudden changes in the motion create no problem with the results.

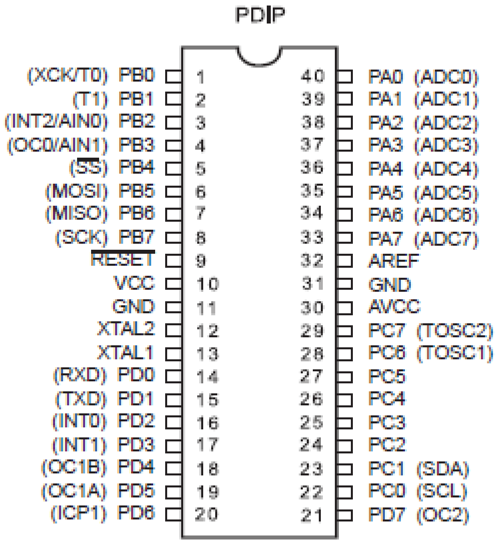
Also, the Robot used for tracking of the suspect is just a prototype and can be actually made into a smaller one (spy robot) so that it cannot be recognized by the suspect. The entire system comprising of software as well as hardware can easily be turned into a completely wireless one.

8. FUTURE SCOPE

1. We had implemented our project using microcontroller 8535 but DSP processor could have been used for fast action and better processing.
2. The robot wheels have been designed to travel on plain surface only; but it can be modified to travel in presence of obstacles and even climb stair.
3. We have used intex night vision webcam having 2.0 megapixel resolutions hence we could use high definition cameras for better results.
4. Tracking of person have been implemented using Kalman as well as DWT technique but we could use Weiner or core kalman to provide better implementation.
5. We have used MATLAB software for detection and tracking but we could use C or C++ or visual C for fast processing operation.
6. Bluetooth device have been used for interfacing SODT system with microcontroller using ESD100 whose range is limited to approx 10m but an extension option could be provided to increase the range upto 300m.
7. The code could have been further modified or improved to provide better results even for low contrast images taken in low lighting conditions.

APPENDIX

- **Microcontroller I.C (8535)**



The ATmega8535 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing instructions in a single clock cycle, the ATmega8535 achieves throughputs approaching 1 MIPS per MHz allowing the system designed to optimize power consumption versus processing speed.

The AVR core combines a rich instruction set with 32 general purpose working registers. All 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8535 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, In- Circuit Emulators, and evaluation kits. This section describes the different memories in the ATmega8535. The AVR architecture has two main memory spaces, the Data Memory and the Program Memory space. In addition, the ATmega8535 features an EEPROM Memory for data storage. All three memory spaces are linear and regular.

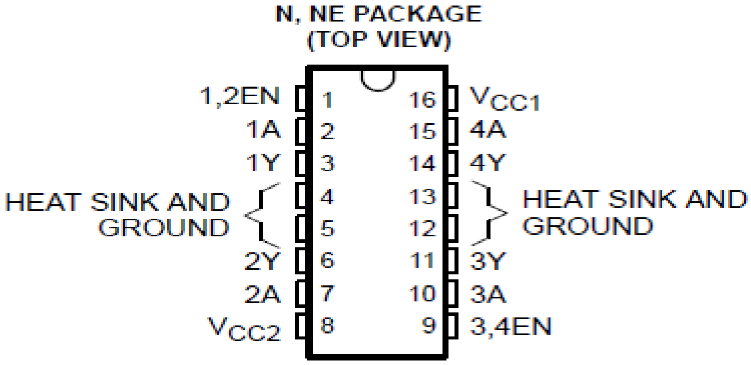
The ATmega8535 contains 8K bytes On-chip In-System Reprogrammable Flash memory for program storage. Since all AVR instructions are 16 or 32 bits wide, the Flash is organized as 4K x 16. For software security, the Flash Program memory space is divided into two sections, Boot Program section and Application Program section.

Features

• High-performance, Low-power AVR® 8-bit Microcontroller
• Advanced RISC Architecture
 – 130 Powerful Instructions
 – Most Single Clock Cycle Execution
 – 32 x 8 General Purpose Working Registers
 – Fully Static Operation
 – Up to 16 MIPS Throughput at 16 MHz
 – On-chip 2-cycle Multiplier

• Nonvolatile Program and Data Memories
– 8K Bytes of In-System Self-Programmable Flash
Endurance: 10,000 Write/Erase Cycles
– Optional Boot Code Section with Independent Lock Bits
In-System Programming by On-chip Boot Program
True Read-While-Write Operation
– 512 Bytes EEPROM
Endurance: 100,000 Write/Erase Cycles
– 512 Bytes Internal SRAM
– Programming Lock for Software Security
• Peripheral Features
– Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
– One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
– Real Time Counter with Separate Oscillator
– Four PWM Channels
– 8-channel, 10-bit ADC
8 Single-ended Channels
7 Differential Channels for TQFP Package Only
2 Differential Channels with Programmable Gain at 1x, 10x, or 200x for TQFP
Package Only
– Byte-oriented Two-wire Serial Interface
– Programmable Serial USART
– Master/Slave SPI Serial Interface
– Programmable Watchdog Timer with Separate On-chip Oscillator
– On-chip Analog Comparator
• Special Microcontroller Features
– Power-on Reset and Programmable Brown-out Detection
– Internal Calibrated RC Oscillator
– External and Internal Interrupt Sources
– Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and extended Standby
• I/O and Packages
– 32 Programmable I/O Lines
– 40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad QFN/MLF
• Operating Voltages

- **L293D**



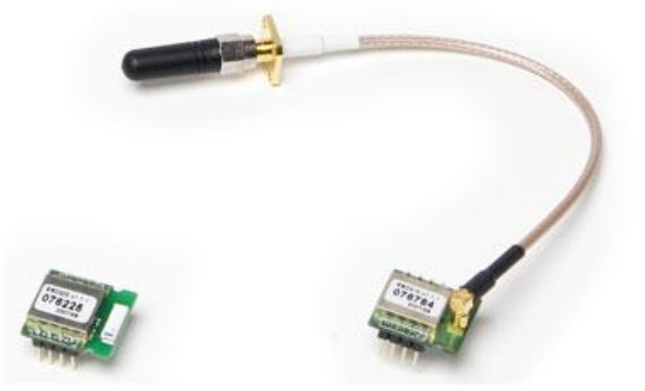
Featuring Unitrode L293 and L293D
Products Now From Texas Instruments

- Wide Supply-Voltage Range: 4.5 V to 36 V
- Separate Input-Logic Supply
- Internal ESD Protection
- Thermal Shutdown
- High-Noise-Immunity Inputs
- Functional Replacements for SGS L293 and SGS L293D
- Output Current 1 A Per Channel (600 mA for L293D)
- Peak Output Current 2 A Per Channel (1.2 A for L293D)
- Output Clamp Diodes for Inductive

Description

The L293 and L293D are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high current/high-voltage loads in positive-supply applications. All inputs are TTL compatible. Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN.
When an enable input is high, the associated drivers are enabled and their outputs are active and in phase with their inputs. When the enable input is low, those drivers are disabled and their outputs are off and in the high-impedance state. With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications. On the L293, external high-speed output clamp diodes should be used for inductive transient suppression. A VCC1 terminal, separate from VCC2, is provided for the logic inputs to minimize device power dissipation. The L293and L293D is characterized for operation from 0^0C to 70^0C.

- **ESD100**



The Parani-ESD200 and Parani-ESD210 are Class 2 type of Compact Embedded Bluetooth Serial Modules for RS-232 cable replacement. By default, they support 30 meters of wireless transmit distance. The Parani-ESD210 has an extension option so that users can extend the transmit distance up to 300 meters using optional antennas.

- Output Interface UART, Compliant Bluetooth Specification v1.2- improved AFH(Adaptive Frequency Hoping), Fast connection
- Transmit Power - ESD200/210: Max. +4dBm
- Receiving Sensitivity - ESD200/210 : -80dBm(0.1%BER)
- Compact size - ESD200/210: 18 x 20 x 11.7 (mm)
- Provides transparent RS232 serial cable replacement.
- Supports Bluetooth Serial Port Profile.
- Interoperability with PDA, laptops etc.
- Built-in chip antenna included
- Supports firmware upgrade via windows-based software(Parani Updater)
- Working distance (In an open field):
  Parani-ESD200: Class 2, Nom. 30meters
  Parani-ESD210: Class 2, Nom. 30meters, up to 300m using patch antenna
- Easy to use Windows configuration tool available.
- No external drivers required.

- **Dongle**

Bluetooth connect low-cost wireless communications and networking between PCs mobile phones & other devices. Bluetooth dongle enable short range wireless data-connectivity between

Bluetooth enabled devices 7 computers. Cellular phones, personal digital assistants (PDA) etc can be connected wirelessly to your computer.

Bluetooth dongle technology enables your PC or notebook for point to multi-point friendly connectivity with other bluetooth devices, such as PDA, mobile phone. Bluetooth is an open specification for wireless data transmission which operates on the globally available 2.4GHz radio frequency.

Bluetooth devices enable communication with each other within a range of approximately 30 feet.

Wirelessly connect your Mobile, PDA, Bluetooth Headsets other bluetooth devices to your PC

Specifications

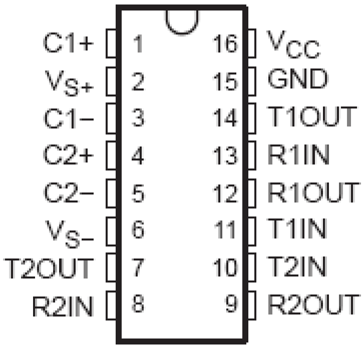
- Support networking, Dial-up, Fax, LAN access, and headset.
- Operation system: Windows 98, 98se, Me, 2000, XP & Vista
- Wireless connect to bluetooth devices such as mobile phone, PDA or PC, for data transfer
- Supporting bluetooth voice data
- Supporting multi languages
- Interface: Compliant with USB 2.0 & 1.1
- Data transmission rate 3 Mb/s

- **Webcam**

Technical Specifications

Features	Description
Model No	IT-305WC
Image sensor	1/7" CMOS sensor
Image Resolution	640x480, 352x288, 320x240, 176x144, 160x120
Frame Rate	Up to 30 fps
Image Control	Brightness, contrast, hue, saturation, gamma, edge ratio
Image Flip	Horizontal, vertical
Monitor Type	CRT, LCD
Environment	Indoor, Outdoor
Focus Distance	4cm ~infinity
Lens View angle	54 degree
I/O interface	USB 1.1,2.0
Image Format	RGB 24, I420
Power Consumption	160mW typical
Operating system	Windows 98/ 2000 / ME / XP

- **MAX232**



Meets or Exceeds TIA/EIA-232-F and ITU
Recommendation V.28
_ Operates from a Single 5-V Power Supply With 1.0-_F Charge-Pump Capacitors
_ Operates up To 120 Kbit/s
_ Two Drivers and Two Receivers
_ Low Supply Current 8mA Typical
_ ESD Protection Exceeds JESD 22
− 2000-V Human-Body Model (A114-A)
_ Upgrade with Improved ESD (15-kV HBM) and 0.1-_F Charge-Pump Capacitors is Available with the MAX202
_ Applications
− TIA/EIA-232-F, Battery-Powered Systems, Terminals, Modems, and Computers

Description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232 F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept 30-V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments Lin ASIC library.

REFERENCES

[1] Alexander C. Berg, Tamara L. Berg, Jitendra Malik, Shape Matching and Object Recognition using Low Distortion Correspondence, IEEE Computer Vision and Pattern Recognition (CVPR) 2005

[2] Multiple description video coding for scalable and robust transmission over IP, Franchi, N. Fumagalli, M. Lancini, R. Tubaro, S. Circuits and Systems for Video Technology, IEEE Transactions on On page(s): 321 - 334, Volume: 15 Issue: 3, March 2005

[3] Portable Preimpact Fall Detector With Inertial Sensors, Ge Wu Shuwan Xue Neural Systems and Rehabilitation Engineering, IEEE Transactions on On page(s): 178 - 183 , Volume: 16 Issue: 2, April 2008

[4] J. H. Tappan, M. E. Wright, and F. E. Sistler, "Error sources in a digital image analysis system", Comput. Electron. Agriculture, vol. 2, pp. 1987

[5] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid, Groups of Adjacent Contour Segments for Object Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, (January 2008)

[6] Independent component analysis of Gabor features for face recognition, Chengjun Liu Wechsler, H. Neural Networks, IEEE Transactions on On page(s): 919 - 928 , Volume: 14 Issue: 4, July 2003

[7] H. Ling and D.W. Jacobs, Using the Inner-Distance for Classification of Articulated Shapes, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005

[8] L. Fei-Fei, R. Fergus, and P. Perona. A Bayesian approach to unsupervised One-Shot learning of Object categories. IEEE Inter. Conf. Computer Vision. 2003

[9] O. Boiman and M. Irani, Detecting Irregularities in Images and in Video. IEEE International Conference on Computer Vision (ICCV), Beijing, October 2005

[10] E. Shechtman and M. Irani, Matching Local Self-Similarities across Images and Videos. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2007

[11] Yang Wang and Greg Mori, Max-Margin Hidden Conditional Random Fields for Human Action Recognition, IEEE Computer Vision and Pattern Recognition (CVPR), 2009

[12] Xuming He, Richard Zemel, and Miguel Carreira-Perpinan. "Multiscale conditional random fields for image labeling." IEEE Conference on Computer Vision and Pattern Recognition, 2004

[13] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, June 2006

[14] Fukunaga, K., Hayes, R.R.: Effects of sample size in classifier design. IEEE Trans. Pattern Anal. Machine Intell. **11**(8), 873–885 (1989)

[15] Doermann, D., Mihalcik, D.: Tools and techniques for video performance evaluation. In: Proceedings of the International Conference on Pattern Recognition, vol. 4, pp. 4167–4170 (2000)

[16] Hua, X.-S., Wenyin, L., Zhang, H.-J.: An automatic performance evaluation protocol for video text detection algorithms. IEEE Trans. Circuits Syst. Video Technol. **14**(4), 498–507 (2004)

[17] Mariano, V.Y., Min, J., Park, J.-H., Kasturi, R., Mihalcik, D., Li, H., Doermann, D., Drayer, T.: Performance evaluation of object detection algorithms. In: Proceedings of the International Conference on Pattern Recognition, vol. 3, pp. 965–969 (2002)

[18] Nagy, G.: Candide's practical principles of experimental pattern recognition. IEEE Trans. Pattern Anal. Machine Intell. **5**(2), 199–200 (1983)

[19] Taylor, G.W.,Wolf, C.: Reinforcement learning for parameter control of text detection in images and video sequences. In: Proceedings of the International Conference on Information & Communication Technologies (IEEE), 2004. IEEE Section France (2004)

[20] Wagner, R.A., Fisher, M.J.: The string to string correction problem. J. Assoc. Comp. Mach. **21**(1), 168–173 (1974)

[21] Brunelli, R. and Poggio, T. (1993). Face recognition: Features versus templates. IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(10):1042–1052

[22] Wolf, C.: Text Detection in Images taken from Videos Sequences for Semantic Indexing. PhD thesis, INSA de Lyon, 20, rue Albert Einstein, 69621 Villeurbanne Cedex, France (2003)

[23] Darrell, T., Essa, I., and Pentland, A. (1996). Task-specific gesture analysis in real-time using interpolated views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12):1236–1242

[24] Turk, M. and Pentland, A. (1992). Face recognition using eigenfaces. In Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 586–590, Hawai

[25] B.Zheng, Y. H. Chang, X. H. Wang, W. F. Good, and D. Gur, ''Feature selection for computerized mass detection in digitized mammograms by using a genetic algorithm,'' Acad. Radiol

[26] J. V. Hajnal, D. L. G. Hill, and D. J. Hawkes, *Medical Image Registration* ~CRC, Boca Raton, FL, 2000

[27] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, ''Multimodal image registration by maximization of mutual information,'' IEEE Trans. Med. Imaging **16**, 187–198 ~1997