VIDEO PROFILING FOR ANOMALY DETECTION

A Dissertation submitted towards the partial fulfillment of the requirement for the award of degree of

Master of Technology
in
Microwave & Optical Communication

Submitted by

SAKSHI SINGLA
2K15/MOC/17

Under the supervision of

Dr.RAJIV KAPOOR
(Professor, Department of ECE)

Department of Electronics & Communication Engineering
Delhi Technological University
(Formerly Delhi College of Engineering)
Delhi-110042
2015-2017

DELHI TECHNOLOGICAL UNIVERSITY

Established by Govt. Of Delhi vide Act 6 of 2009

(Formerly Delhi College of Engineering)

SHAHBAD DAULATPUR, BAWANA ROAD, DELHI-110042

CERTIFICATE

This is to certify that the dissertation title “**VIDEO PROFILING FOR ANOMALY DETECTION**” submitted by **Ms. SAKSHI SINGLA (Roll. No.:- 2K15/MOC/17**, in partial fulfilment for the award of degree of Master of Technology in “**MICROWAVE & OPTICAL COMMUNICATION)”,** run by Department of Electronics & Communication Engineering in Delhi Technological University during the year 2015-2017., is a bonafide record of student’s own work carried out by her, under my supervision and guidance in the academic session 2016-17. To the best of my belief and knowledge, the matter embodied in dissertation has not been submitted for the award of any other degree or certificate in this or any other university or institute.

Prof. RAJIV KAPOOR
Supervisor
Professor (ECE)
Delhi Technological University
Delhi-110042

DECLARATION

I hereby declare that all the information in this document has been obtained and presented in accordance with academic rules and ethical conduct. This report is my own work to the best of my belief and knowledge. I have fully cited all material by others which I have used in my work. It is being submitted for the degree of Master of Technology in Microwave & Optical Engineering at the Delhi Technological University. To the best of my belief and knowledge it has not been submitted before for any degree or examination in any other university.

SAKSHI SINGLA

Date: 17TH JULY, 2017.

M.Tech (MOC)

Place: Delhi Technological University, Delhi

2K15/MOC/17

ACKNOWLEDGEMENT

I owe my gratitude to all the people who have helped me in this dissertation work and who have made my postgraduate college experience one of the most special periods of my life.

Firstly, I would like to express my deepest gratitude to my supervisor **Prof. RAJIV KAPOOR**, **Professor (ECE)** for his invaluable support, guidance, motivation and encouragement throughout the period during which this work was carried out. I am deeply grateful to **DR. S. INDU,H.O.D. (Department Of ECE)**for their support and encouragement in carrying out this project.

I also wish to express my heart full thanks to my classmates as well as staff at Department of Electronics & Communication Engineering of Delhi Technological University for their goodwill and support that helped me a lot in successful completion of this project.

Finally, I want to thank my parents, family and friends for always believing in my abilities and showering their invaluable love and support.

SAKSHI SINGLA
M. Tech. (MOC)
2K15/MOC/17

ABSTRACT

There has been an increasing demand for automatic methods which can analyse huge quantities of video data for surveillance which is continuously generated by closed-circuit television (CCTV) Systems. Our main objective of deploying an automated visual surveillance system is to detect abnormal behavior patterns and recognize the normal ones.There are many other methods which solve the problem of anomaly detection in video surveillance.In this thesis, we present a novel method to detect anomalies in a video surveillance system. We will make use of FEM toolbox in order to detect abnormalities in a video sequence. This toolbox will give us modal frequencies of the frames of a video, which will help us to find out as to which actions are periodic and which actions are non-periodic. Also, we will compare periodicities of various periodic actions and compare and contrast two similar actions on grounds of these frequencies.

Contents

LIST of Figures

LIST of Tables

CHAPTER 1 INTRODUCTION

Abnormal behavior detection has been one of the most important research branches in intelligent video content analysis. Research in abnormal detection has made great progress in recent years, such as abnormal action detection, abnormal event detection and abnormal crowd detection.

There has been an increasing demand for automatic methods which can analyse huge quantities of video data for surveillance which is continuously generated by closed-circuit television (CCTV) Systems. Our main objective of deploying an automated visual surveillance system is to detect abnormal behaviour patterns and recognize the normal ones. To successfully achieve this, we analyse and profile previously observed behavior patterns, and then we develop a criteria as to what is normal and what is abnormal and further apply them to newly captured patterns for anomaly detection.

Due to the decrease in costs of video surveillance equipments, we have ended up with extremely huge amounts of video data. This excessive volume of information is almost impossible to be dealt by human operators.Due to this problem of large amount of surveillance video data to be analyzed and the real-time nature of a lot of surveillance applications, it is a need of the hour to have an automated system that runs in real time and accomplishes the task with little human intervention.

In this thesis, we present a novel method to detect anomalies in a video surveillance system. We will make use of FEM toolbox in order to detect abnormalities in a video sequence.

This toolbox will give us modal frequencies of the frames of a video, which will help us to find out as to which actions are periodic and which actions are non periodic. Also, we will compare periodicities of various periodic actions and compare and contrast two similar actions on grounds of these frequencies.

The above mentioned toolbox runs on MATLAB and so the entire results of this thesis are worked out on MATLAB.

The rest of the thesis is organised as follows. Chapter 2 gives a literature review of various anomaly detection techniques that have been put to use. Chapter 3 and Chapter 4 give us some insight on Video Profiling & Anomaly Detection. Chapter 5 discusses about the method FEM and its toolbox in MATLAB. In Chapter 6 and Chapter 7, we have presented our proposed methodology and its results on implementation. Finally, conclude the thesis in Chapter 8 followed by references.

CHAPTER 2 LITERATURE REVIEW

The process of anomaly detection generally comprises of two different phases: a training phase and a testing phase. In the training phase, the general traffic profile is defined and in the testing phase, the learned profile is applied to the new data.

2.1 PREMISE OF ANOMALY DETECTION

The actual premise i.e. main approach for anomaly detection is that the intrusive activity is a subset of anomalous activity. Let us take into account an outsider trying to intrude, who does not have any idea about the actual user's activity patterns, trying to intrude into a host system, there exists a high possibility that the activity of the intruder will be detected as anomalous. In the ideal case, the set of anomalous activities will be the same as the set of intrusive activities. In such a case, flagging all anomalous activities as intrusive activities results in no false positives and no false negatives. However, intrusive activity does not always coincide with anomalous activity. Kumar and Stafford suggested that there are four possibilities, each with a non-zero probability:

- Intrusive but not anomalous: These are false negatives. An intrusion detection system fails to detect this type of activity as the activity is not anomalous. These are called false negatives because the intrusion detection system falsely reports the absence of intrusions.

- Not intrusive but anomalous: These are false positives. In other words, the activity is not intrusive, but because it is anomalous, an intrusion detection system reports it as intrusive. These are called false positives because an intrusion detection system falsely reports intrusions.

- Not intrusive and not anomalous: These are true negatives; the activityis not intrusive and is not reported as intrusive.

- Intrusive and anomalous: These are true positives; the activity is intrusive and is reported as such.

When false negatives need to be minimized, thresholds that define an anomaly are set low. This results in many false positives and reduces the efficacy of automated mechanisms for intrusion detection. It creates additional burdens for the security administrator as well, who must investigate each incident and discard false positive instances.

2.2 STATISTICAL ANOMALY DETECTION

In statistical methods for anomaly detection, the system observes the activity of subjects and generates profiles to represent their behavior. The profiletypically includes such measures as activity intensity measure, audit record distribution measure, categorical measures (the distribution of an activity over categories) and ordinal measure (such as CPU usage). Typically, two profiles are maintained for each subject: the current profile and the stored profile. As the system/network events (viz. audit log records, incoming packets, etc.) are processed, the intrusion detection system updates the current profile and periodically calculates an anomaly score (indicating the degree of irregularity for the specific event) by comparing the current profile with the stored profile using a function of abnormality of all measures within the profile. If the anomaly score is higher than a certain threshold, the intrusion detection system generates an alert.

Statistical approaches to anomaly detection have a number of advantages. Firstly, these systems, like most anomaly detection systems, do not require prior knowledge of security flaws and/or the attacks themselves. As a result, such systems have the capability of detecting ‘‘zero day’’ or the very latest attacks. In addition, statistical approaches can provide accurate notification of malicious activities that typically occur over extended periods of time and are good indicators of impending denial-of-service (DoS) attacks. A very common example of such an activity is a portscan. Typically, the distribution of portscans is highly anomalous in comparison to the usual traffic distribution. This is particularly true when a packet has unusual features (e.g., a crafted packet). With this in mind, even portscans that are distributed over a lengthy time frame will be recorded because they will be inherently anomalous.

However, statistical anomaly detection schemes also have drawbacks. Skilled attackers can train a statistical anomaly detection to accept abnormal behavior as normal. It can also be difficult to determine thresholds that balance the likelihood of false positives with the likelihood of false negatives. In addition, statistical methods need accurate statistical distributions, but, not all behaviors can be modeled using purely statistical methods. In fact, a majority of the proposed statistical anomaly detection techniques require the assumption of a quasi-stationary process, which cannot be assumed for most data processed by anomaly detection systems.

Haystack is one of the earliest examples of a statistical anomaly based intrusion detection system. It used both user and group based anomaly detection strategies, and modeled system parameters as independent, Gaussian random variables. Haystack defined a range of values that were considered normal for each feature. If during a session, a feature fell outside the normal range, the score for the subject was raised. Assuming the features were independent, the probability distribution of the scores was calculated. An alarm was raised if the score was too large. Haystack also maintained a database of user groups and individual profiles. If a user had not previously been detected, a new user profile with minimal capabilities was created using restrictions based on the user’s group membership. It was designed to detect six types of intrusions: attempted break-ins by unauthorized users, masquerade attacks, penetration of the security control system, leakage, DoS attacks and malicious use. One drawback of Haystack was that it was designed to work offine. The attempt to use statistical analyses for real-time intrusion detection systems failed, since doing so required high performance systems. Secondly, because of its dependence on maintaining profiles, a common problem for system administrators was the determination of what attributes were good indicators of intrusive activity.

One of the earliest intrusion detection systems was developed at the Stanford Research Institute (SRI) in the early 1980’s and was called the Intrusion Detection Expert System (IDES). IDES was a system that continuously monitored user behavior and detected suspicious events as they occurred. In IDES, intrusions could be flagged by detecting departures from established normal behaviorpatterns for individual users.As theanalysis methodologiesdeveloped for IDES matured,scientists atSRI developedan improvedversion ofIDEScalled the Next-Generation Intrusion Detection Expert System (NIDES). NIDES was one of the few intrusion detection systems of its generation that could operate in real time for continuous monitoring of user activity or could run in a batch mode for periodic analysis of the audit data. However, the primary mode of operation of NIDES was to run in real time. A flow chart describing the real time operation of NIDES is shown in the below Fig.2. Unlike IDES, which is an anomaly detection system, NIDES is a hybrid system that has an upgraded statistical analysis engine. In both IDES and NIDES, a profile of normal behavior based on a selected set of variables is maintained by the statistical analysis unit. This enables the system to compare the current activity of the user/system/network with the expected values of the audited intrusion detection variables stored in the profile and then flag an anomaly if the audited activity is suffciently far from the expected behavior. Each variable in the stored profile reflects the extent to which a particular type of behavior is similar to the profile built for it under ‘‘normal conditions’’ The way that this is computed is by associating each measure/variable to a corresponding random variable. The frequency distribution is built and updated over time, as more audit records are analyzed. It is computed as an exponential weighted sum with a half-life of 30 days. This implies that the half-life value makes audit records that were gathered 30 days in the past to contribute with half as much weight as recent records; those gathered 60 days in the past contribute one-quarter as much weight, and so on. The frequency distribution is kept in the form of a histogram with probabilities associated with each one of the possible ranges that the variable can take. The cumulative frequency distribution is then built by using the ordered set of bin probabilities. Using this frequency distribution, and the value of the corresponding measure for the current audit record, it is possible to compute a value that reflects how far away from the ‘‘normal’’ value of the measure the current value is. The actual computation in NIDES renders a value that is correlated with how abnormal this measure is. Combining the values obtained for each measure and taking into consideration the correlation between measures, the unit computes an index of how far the current audit record is from the normal state. Records beyond a threshold are flagged as possible intrusions.

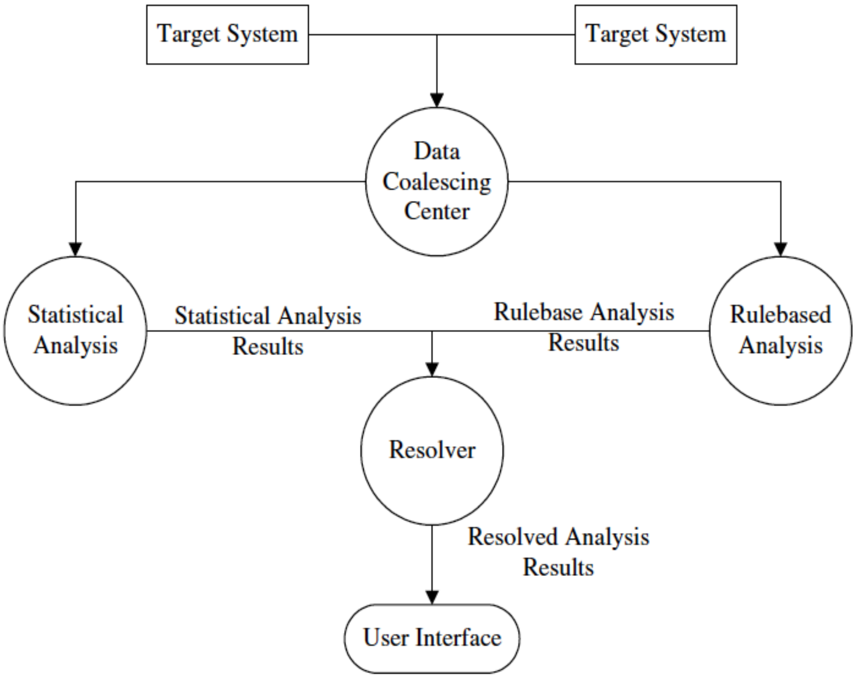


Figure 2.1 - Flow chart of real time operation in NIDES

However the techniques used in have several drawbacks. Firstly,the techniques are sensitive to the normality assumption. If data on a measure are not normally distributed, the techniques would yield a high false alarm rate. Secondly, the techniques are predominantly univariate in that a statistical norm profile is built for only one measure of the activities in a system. However, intrusions often affect multiple measures of activities collectively.

Statistical Packet Anomaly Detection Engine (SPADE) is a statistical anomaly detection system that is available as a plug-in for SNORT, and can be used for automatic detection of stealthy port scans. SPADE was one of the first papers that proposed using the concept of an anomaly score to detect port scans, instead of using the traditional approach of looking at p attempts over q seconds. In this, the authors used a simple frequency based approach, to calculate the ‘‘anomaly score’’ of a packet. The fewer times a given packet was seen, the higher was its anomaly score. In other words, the authors define an anomaly score as the degree of strangeness based on recent past activity. Once the anomaly score crossed a threshold, the packets were forwarded to a correlation engine that was designed to detect port scans. However, the one major drawback for SPADE is that it has a very high false alarm rate. This is due to the fact that SPADE classifies all unseen packets as attacks regardless of whether they are actually intrusions or not.

Anomalies resulting from intrusions may cause deviations on multiple measures in a collective manner rather than through separate manifestations on individual measures. To overcome the latter problem, Ye et al. presented a technique that used the Hotellings T^2 test to analyze the audit trails of activities in an information system and detect host based intrusions. The assumption is that host based intrusions leave trails in the audit data. The advantage of using the Hotellings T^2 test is that it aids in the detection of both counter relationship anomalies as well as mean-shift anomalies. In another paper, Kruegel et al. show that it is possible to find the description of a system that computes a payload byte distribution and combines this information with extracted packet header features. In this approach, the resultant ASCII characters are sorted by frequency and then aggregated into six groups. However, this approach leads to a very coarse classification of the payload.

A problem that many network/system administrators face is the problem of defining, on a global scale, what network/system/user activity can be termed as ‘‘normal’’. Maxion and Feather characterized the normal behavior in a network by using different templates that were derived by taking the standard deviations of Ethernet load and packet count at various periods in time. An observation was declared anomalous if it exceeded the upper bound of a predefined threshold. However, Maxion et al. did not consider the non-stationary nature of network traffic which would have resulted in minor deviations in network traffic to go unnoticed .

More recently, analytical studies on anomaly detection systems were conducted. Lee and Xiang used several information-theoretic measures, such as entropy and information gain, to evaluate the quality of anomaly detection methods, determine system parameters, and build models. These metrics help one to understand the fundamental properties of auditdata.

2.3 ANOMALY DETECTION BY MACHINE LEARNING

Machine learning is known as the field of study which gives computers i.e. program and system to learn without being explicitly programmed and simultaneously improve its performance on a particular group of tasks.Machine learning is aimed to provide solutions to similar questions as provided by statistics or data mining. However, unlike approaches used in statistics which focus on understanding process of data generation, techniques deployed by machine learning concentrate on building program/system which improves on its own performance based on historical results. Thus, systems/applications based on this paradigm have the ability to process newly acquired information and alter their execution strategy.

2.3.1 SEQUENCE ANALYSIS BASED ON SYSTEM CALL.

Most widely used techniques of machine learning for anomaly detection involves knowing about the behavior of a program and then finding out significant deviations from the standard. In one of the seminal papers, Forrest and others did establish an analogy between the human immune system and intrusion detection. This was done by proposing a methodology which involved analyzing call sequences of a program's system to help them build a normal profile. Several UNIX based programs like sendmail, lpr, etc., were analyzed which depicted that correlations in fixed length sequences of the system calls will be helpful in building a normal profile of the program. Therefore, programs which show deviations in the sequences from the normal sequence profile can be then considered as victimized. The system developed by them was initialy used off-line using historical data by using a simple table-lookup algorithm to learn various profiles of programs. Their work was further extended by Hofmeyr and others, where they collected a database of normal behavior for each program / application of interest. After the construction of stable database for a given program in a particular environment, it was thenused tomonitor behavior of the program. The sequences of system calls thus resulted then formed the set of normal patterns for the existing database, and sequences which were not present in the database indicated anomalies.

2.3.2 BAYESIAN NETWORKS

Bayesian network is a graphical model which encodes probabilistic relationships among interest variables. They have several advantages for data analysis when used with statistical techniques.Firstly, they can handle issues of incomplete data because of their ability toencode interdependencies between variables. Secondly, Bayesian networks can be used to predict consequences of various actions as they tend to represent causal relationships. Lastly, these networks are useful in modeling problems where new data needs to be combined with learnings of the model because of their causal and probabilistic relationships. Several researchers with ideas from Bayesian statistics have created models foranomaly detection. Valdes and others have developed an anomaly detection system which employs new Bayesian networks to perform intrusion detection on traffic bursts. Model by Valdes has the potential to detect distributed attacks whereeach single attack session is not suspicious enough that could generate an alert. On the other hand,it also had few disadvantages. One, the classification capability of unlearned Bayesian networks is same as of a threshold based system which calculates the sum of the outputs based on the inputs from the child nodes. Also, including additional information become strenuous as variables possessing information cannot interact directly with the child nodes, and they only affect the probability of root nodes.

Another area, within the field of anomaly detection, which frequently use Bayesian techniques is classification and repression of false alarms. Kruegel and others proposed a multi-sensor fusion approach where a single alarm was produced by the aggregation of outputs of different IDS sensors.This method is based on the premise that any anomaly detection technique cannot confidently classify a set of events as an intrusion. Although using Bayesian networks for intrusion detection or intruder behavior prediction might be effective in various applications, their limitations should be kept in mind during actual implementation. As there are various beliefs on the behavioral model of target system that determine the credibility of the method, deviating from those will reduce its credibility. Selection of faulty models might lead to an erroneous detection system. Thus, selection of an accurate model is the initial step towards solving the problem which unfortunately is not an easy task due to complex nature of systems and networks.

2.3.3 PRINCIPAL COMPONENTS ANALYSIS

Intrusion detection is done typically on extensive and multidimensional datasets. With the advent of high-speed networks and data intensive application based on distributed network have made storing, processing, transmitting, visualizing and understanding data more expensive and tricky. To overcome this problem of high dimensional datasets, researchers have tried developing techniques to reduce dimensions and have led to methods known as principal component analysis(PCA). In mathematical terms, it is a technique where 'n' correlated random variables are transformed into $d \leq n$ uncorrelated variables, which arenothing but the linear combinations of the original variables & might be used to dsiplay reduced form of data. Generally, first principal component of the transformation is the linear combination of the original variables possessinglargest variance. Thus, first principal component is the projection on the direction in which the variance of the projection is maximized. 2nd principal component is linear combination of the original variables with 2nd largest variance & is orthogonal to the first set of principal components, and so on. In many data sets, the first several principal components contribute most of the variance in the original dataset, so that the remaining components can be disregarded with minimal loss of thevariance for dimension reduction of the dataset.

Major usage of PCA technique has been in the domain of pattern recognition, image compression, andintrusion detection. Shyu and others proposed an anomaly detectionscheme, in which PCA was used as an outlier detector and was then applied to reduce the dimensionality of the audit data to arrive at a classifier which is a function of the principal components. For anomaly detection, Mahalanobis distance, calculated based on sum of squares of standarized principal component scores,of each observation from the center was measured. Shyu and others evaluated their method over the KDD CUP 99 data & haveshown that it exhibits superior detection rates than other outlier basedanomaly detection algorithms like Local Outlier Factor ''LOF'', Nearest Neighbor &Kth Nearest Neighbor approach. Other well-known techniques that employ PCA methodology include the work done by Wang, Bouzida and others.

2.3.4 MARKOV MODELS

Markov chains, have also been employed extensively for anomaly detection. Ye and others, present an anomaly detection technique based on Markov chains and they have studied system call event sequences from the recent past by opening an observation window of size N. The type of audit events,

$$E_{t-N+1}, \dots \dots \dots \dots, E_t$$

in the window at time t was examined and the sequence of states

$$X_{t-N+1}, \dots \dots \dots \dots, X_t$$

obtained. Subsequently, the probability that the sequence of states

$$X_{t-N+1}, \dots \dots \dots \dots, X_t$$

is normal was obtained. The larger the probability, the more likely the sequence of states results from normal activities. A sequence of states from attack activities is presumed to receive a low probability of support from theMarkov chain model of the normal profile.

Hidden Markov model (HM model), another popular Markov technique, is a statistical model where the system being modeled is assumed to be a Markov process with unknown parameters. The challenge is to find out the hidden para meters from thegiven observable parameters. It is different from a regular Markov model in a way as the elements visible are variables of the system, being influenced by the state of the system, which is itself hidden. In regular model, only parameters are state transition probabilities and also system's state is directly observable. States of the HM represent indistinct conditions of the system currently being modeled. In each state, observable system outputs are produced with a certain probability and another probability of likely next state is also displayed. Model can thus represent non-stationary sequences due to different output probability distributions in each state, and allowing for change of system state with time.

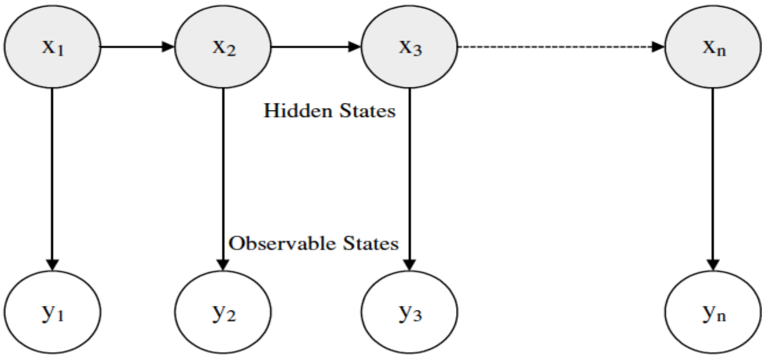


Figure 2.2 - Hidden Markov Model

To estimate the parameters of a HM model for modeling normal system behavior, sequences of normal events collected from normal system operation are used astraining data. To estimate theparameters of HM, expectation-maximization (EM)algorithm is used. Aftertraining of HM model, test data when encountered results in probability measures which can be used as thresholds for anomaly detection. Three key problems are faced while using HMM for anomaly detection. First, the evaluation problem, is to determine the probability of observed sequence being generated by the model given the sequence of observations. Second is the learning problem which involves building a model, or a set of models, from a set off data which is in sync with the observed behavior. Given HM model & its associated observations, third problem, known as the decoding problem, involves determining the most likely set of hidden states that have led to those observations.

Warrender and others compared the performance of four methods viz., (i) simple enumeration of observed sequences, (ii) comparison of relative frequencies of different sequences, (iii) a rule induction technique& (iv) HM models at representing normal behavior accurately and recognizing intrusions in system call datasets. Warrender shows that while HM models outperform the other three methods, it comes at a greater computational cost. In the model that is put forward, the authors use HM model with fully connected states such that transitions from any state to any other state are allowed. Thus, any process that issues S system-calls will have S-states, which implies that roughly there will be $2S^2$ values in the state transition matrix (STM). In a computer network/system, a large number of system calls are issued by the process. Thus, modeling every process in a computer system/network would become computationally infeasible.

In another paper, Yeung described the use of HM models for anomaly detection based on profiling system, call sequences and the shell command sequences. Once the model is expereinced, it computes the sample likelihood of the observed sequence using either forward or backward algorithm. Probability threshold, on the basis of minimum likelihood among all training sequences, was deployed to differentiate a normal behavior from ananomalous behavior. Major limitation of this particular approach is that it lacked generalization & support for those users which are not uniquely identified by thecurrent system.

Mahoney and other co-authors came up with various methods as solution to problems incurred while detecting anomalies in the usage of network protocols by observing packet headers. The shared factor among all of them is the structured application of learning techniques which helps in obtaining profiles automatically of normal behavior for protocols at distinct layers. Mahoney experimented with anomaly detection by range matching network packet header fields over the DARPA network data. Packet Header Anomaly Detector (PHAD), Learning Rules for Anomaly detection (LeRAD) and Application Layer Anomaly Detector (ALAD) all of them use time-based models in which the probability of an event depends on its last occurence. For each aspect, they garner a set of allowed values and flag new values as anomalous.

PHAD, ALAD, and LeRAD differ in the aspects that they observe as shown below. (a) PHAD monitors 33 attributes from ethernet, IP and transport layer packet headers. (b) ALAD models incoming server TCP requests: source and destination IP addresses and ports; opening and closing TCP flags; and the list of commands in the application payload. It builds separate models for each target host, port number (service), or host/port combination on the basis of each attribute. (c) LeRAD also models TCP connections. Though, the data set is a multivariate network traffic data containing fields extracted out of the packet headers, the author tries to break it down into a set of univariate problems and then sum the weighted results along each dimension from range matching. This method results in more computationally efficient technique in addition to the effective detection of network intrusions, though breaking multivariate data into univariate data leads to in-neffective detection. For example, in a typical SYN flood attack an indicator of the attack, will have more SYN requests than usual, but will observe a lower than normal ACK rate. As higher SYN rate or lower ACK rate alone can both happen during normal usage, it is the combination of both higher SYN rate and lower ACK rate that signals the attack.

Major demerit of many of these machine learning techniques, like system call based sequence analysis approach or HM model approach is that they are resource expensive. E.g. ananomaly detection technique based on the Markov chain model is computationally expensive because it uses parametric estimation techniques based on the Bayes' algorithm for learning the normal profile of the host/network under consideration. If large amount of audit data is taken into consideration with relatively such high frequency of events they occur is systems today, a technique like this for anomaly detection is not feasible for real time operation.

2.4 UNUSUAL EVENT DETECTION IN HUMAN CROWDS USING OPTICAL FLOW

Investigating human task is an essential issue in video surveillance and is a testing task because of their inclination towards non-rigid shapes. In this paper, optical flows are first assessed and afterward utilized for an intimation to a bunch of human crowds into bunches in an unsupervised way utilizing our proposed strategy for adjacency-matrix based clustering (AMC). While we get theclusters of human crowds, their behaviors with characteristics, orientation, position and group size, are portrayed by a model of force field. At last, we can foresee the behaviors of human crowds in light of the model and after that recognize if any oddities of human crowd(s) are exhibited in the scene. Experimental results acquired by utilizing broad dataset demonstrate that our above mentioned framework is powerful in distinguishing abnormal occasions for uncontrolled environment of surveillance videos.

Thus, in this work, optical flows are first evaluated and after that are utilized for a hint to cluster human crowds into bunches in an unsupervised way utilizing our proposed clustering1method. While theclusters of human group are found out, their features, for example, orientation, position and crowd size, are characterised by a model of force field. At last, we can foresee the behavior of human crowd in view of the model and afterward distinguish if any abnormalities of human crowd(s) are there in the scene.

2.5 ENERGY MODEL APPROACH FOR ABNORMAL CROWD BEHAVIOR DETECTION

Abnormal crowd behavior identification has a fundamental impact in surveillance tasks. We put forward a camera parameter which is autonomous and perspective distortion invariant strategy to manage distinguishing two sorts of unusual group behaviors. The two common abnormal activities are people gathering and running. Because checking is basic for recognizing theanomalous group behavior, we present a potential energy based model to gauge the quantity of people out in general society. Constructing histograms on the X-and Y-axis, independently, we can secure probability distribution of the forefront object and a later describe entropy of crowd. We present the Crowd Distribution Index by merging the overall population count with entropy of crowd to symbolize the spatial distribution of the group. We set a lower restrain on Crowd Distribution Index to recognize people's gathering. To recognize running people, the kinetic energy is discovered by estimation of optical flow and Crowd Distribution Index. With a limit, motor vitality can be used to recognize running people. To test the performance of our paradigm, videos of various scenes and variouscrowd densities are used as a part of our investigations. Without camera change and training data, this procedure can precisely recognize abnormal behavior with low calculation load.

The framework comprises of two methodology: people counting and abnormal crowd behavior detection, as parts A and B appeared in the figure. The yield of section A is the contribution in part B. In the algorithms used for counting the number of people, we expect that individuals move on a known ground plane where there are no other moving items, for example, automobiles or other creatures. The yield of section A are foreground pixels and people count. Like most existing human counting calculations, we utilize a versatile Gaussian Mixture Modeling (GMM) strategy to extract the foreground from videos, and reject commotion by applying a binary morphology method. At that point an underlying model for counting the people is examined. From the pinhole point of view projection, we have two perceptions. In light of the two perceptions, we introduce a picture potential energy model to enhance the underlying model. At last, we exhibit the strategy for parameter estimation which is important to the picture potential energy model. Part B aims to distinguish two regular anomalous group behaviors: individuals assembling and running. We outline the detection algorithm Based on color image processing, we propose a programmed bi-directional counting strategy with one color video hung from the roof of the door. By utilizing Markov random field (MRF), we display a calculation for individuals counting. Three sorts of picture features which meet the Markov properties are removed. At that point a least square strategy is implemented to evaluate the general population number. In related works, a joined human segmentation and group tracking technique for individuals counting is displayed. Also, a model-indicated directional filter (MDF) is utilized to recognize object candidate areas taken after by a novel matching procedure to distinguish the person on foot head positions and the number of individuals can be tallied from the quantity of the heads. Zhao et al. portion the frontal area blobs with someearlier learning of human shape in a Bayesian system. Liu et al. use a technique for camera auto-alignment in view of information gathered by tracking individuals and present a model based division algorithm which partitions a gathering of individuals into individual. The detection based technique can acquire attractive outcomes. Be that as it may, edge extraction and tracking is a tedious procedure. At the point when the environment is convoluted or comprising up to 10 individuals, the detection based techniques dependably come up short due to high crowd density, occlusions and high computational load. Consequently, these frameworks are of constrained utilization in urban environments, which frequently contain vast groups of individuals with serious occlusions.

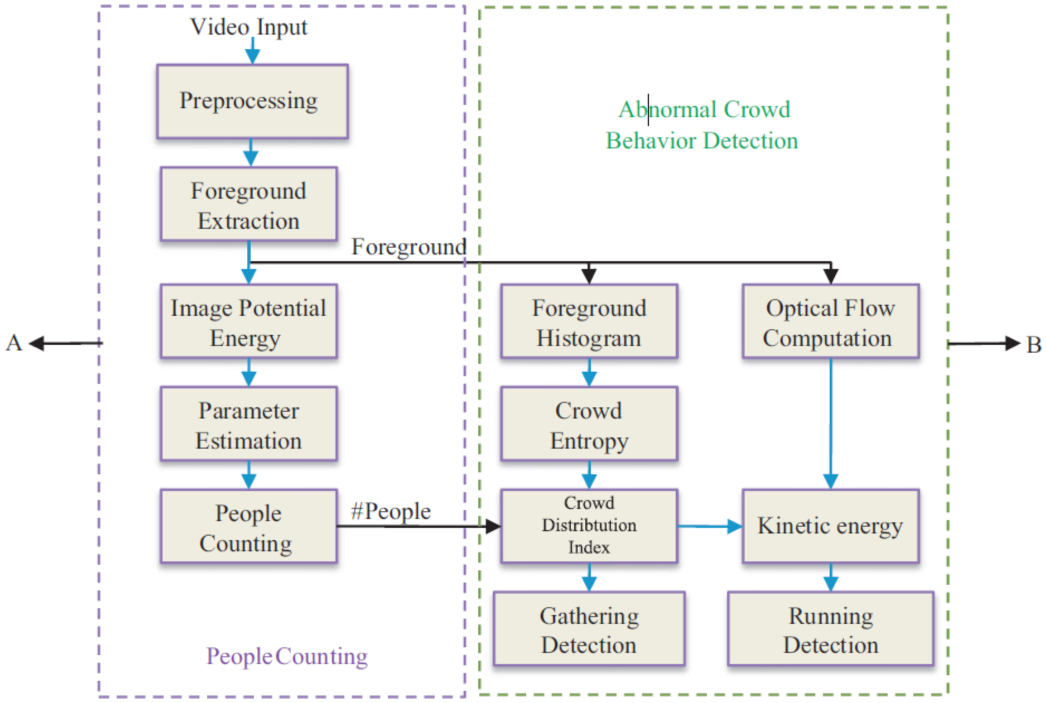


Figure 2.3 - System Architecture

2.6 ANOMALY DETECTION BASED ON TRAJECTORY SPARSE RECONSTRUCTION ANALYSIS

Abnormal behavior detection has been a standout amongst the most imperative research branches for intelligent video content investigation. Here, we put forward an abnormal behavior detection strategy by presenting trajectory sparse reconstruction analysis (SRA). Given a video situation, we gather trajectories of normal behaviors and concentrate the control point features of cubic B-spline curves to develop a typical normal dictionary , that is additionally divided into Route sets. In the dictionary set, sparse reconstruction coefficients and residuals of a test trajectory to the Route sets can also be figured by SRA . The minimal residual is utilized to arrange the test behavior into an ordinary behavior or an anomalous one. SRA is solved by L1-norm minimization, prompting that a couple of dictionary tests are utilized while reconstructing behavior trajectory , which ensures that the approach put forward is substantial notwithstanding when the dictionary set is small. Experimental outcomes with comparisons demonstrate that the proposed method enhances the best in class method.

An outline of the proposed method in this paper, in light of a presettrajectory dictionary set for a set video situation, we put forward a novel abnormal behavior recognition strategy utilizing sparse reconstruction analysis. We initially gather a set of trajectories of normal behaviors from video by an object tracking algo or a movement detection technique. By watching their appearances, thesetrajectories are physically classified into various subsets, known asRoute sets. For the entire gathered trajectories , the Least-squares Cubic Spline Curves Approximation (LCSCA) highlights are extricated for portrayals and after that development of thedictionary set. When performing anomalous behavior detection, each test trajectory will likewise be symbolized with LCSCA highlights. At that point, we present thesparse reconstruction analysis on the normal dictionary set to order the testing movement trajectories of objects, where our goal is to recreate the test trajectory with as few dictionary samples as we can. The L1-norm minimization is utilized to comprehend the reconstruction coefficients, on which the reconstruction residuals of each Route set can likewise be figured. The minimal reconstruction residual is utilized to order the test trajectory into a normal behavior or an anomalous one with an empirically characterized threshold. The structure of the proposed approach is shown in the figure below.

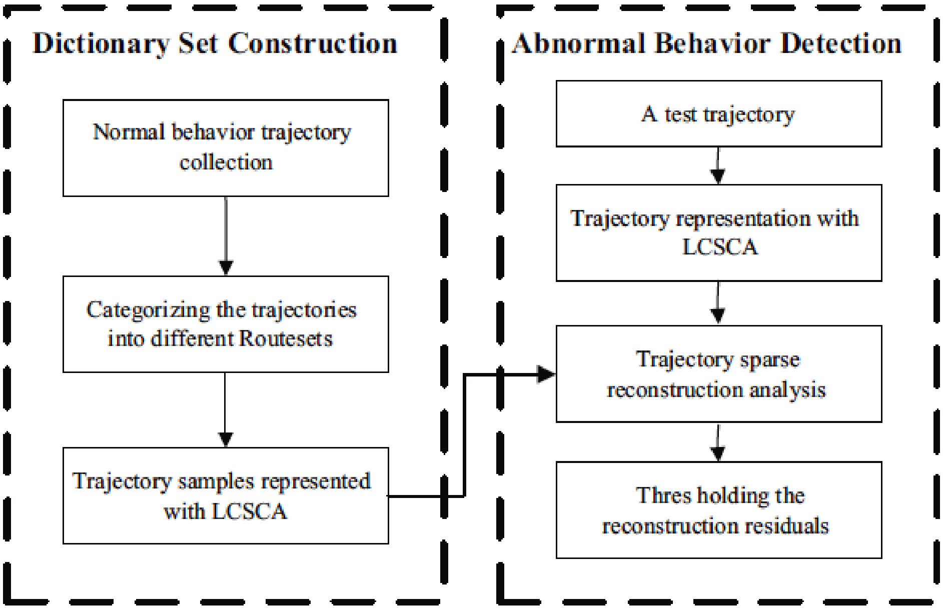

Dictionary Set Construction
Normal behavior trajectory collection
Categorizing the trajectories into different Routesets
Trajectory samples represented with LCSCA
Abnormal Behavior Detection
A test trajectory
Trajectory representation with LCSCA
Trajectory sparse reconstruction analysis
Thres holding the reconstruction residuals

Figure 2.4 - Framework of SRA approach

Experimental outcomes on a real-world dataset demonstrate the great execution of our proposed approach on various sizes of samples. The comparison with the conventional approach is likewise given, which demonstrates that the proposed approach accomplishes a better outcome despite the fact that a smaller dictionary set is utilized. A known detriment of the proposed technique is that the detection efficiency is influenced by the control point parameter, which ought to be enhanced later in future work.

CHAPTER 3 VIDEO BEHAVIOR PROFILING

Let us now explain the problem of behavior profiling. Profiling basically means the procedure of extrapolating some information about a person or an entity based on it's known tendencies or traits. Similarly, here we define our problem of behavior profiling on the basis of the following.

Let us consider a training data set D which consists of N feature vectors

$$D = \{P_1,...,P_2,...,P_n,...,P_N\}$$

where P_n is defined as

$$P_n = [p_{n1},...,p_{nt},...,p_{nTn}]$$

that represents the pattern of behavior which is captured by v_n, captured by the nth video segment. The issue that we are addressing here is that we need to discover the natural grouping of our training behavior patterns on which we can construct a model for normal behavior. This is basically a data clustering problem where the number of clusters is unfamiliar. There are a lot of factors that can make this problem difficult:

1. The length of every feature vector P_n can be distinct. Traditional clustering approaches like mixture models and K-means have a condition that each and every data sample should be symbolized as a feature vector of set length. Therefore these methods can't be implemented straight away .
2. Among these variable length highlight vectors, the meaning of a distance/affinity metric is nontrivial. Measuring partiality amid include vectors of variable length frequently includes Dynamic Time Warping (DTW). A standard DTW technique utilized as a part of PC vision group would try to regard the feature vector P_n as a K_e-dimensional trajectory and measure the separation of 2 behavior patterns by discovering correspondence between discrete vertices on two trajectories. Because in our system, a behavior pattern is spoken to as an arrangement of temporal correlated events, i.e., a stochastic procedure, a stochastic model-

based approach is more proper for measuring the distance. Note that on account of matching two arrangements of various lengths in view of video object recognition, the affinity of the most comparative pair of pictures from two sequences can be utilized for sequence affinity measurement. Nonetheless, because we concentrate on the modeling behavior that can include various objects interacting over space and time, the above mentioned method can't be connected straightforwardly for this situation.

3. Model choice should be performed to decide the quantity of clusters. To conquer the previously mentioned problem, we put forward a spectral clustering algorithm with feature and model choice in view of modeling every behavior pattern utilizing a Dynamic Bayesian Network. The figure below demonstrates a diagrammatic representation of our behavior profiling approach. It demonstrates unmistakably that the proposed spectral clustering algorithm (blocks inside the dashed box) is the center of our method.

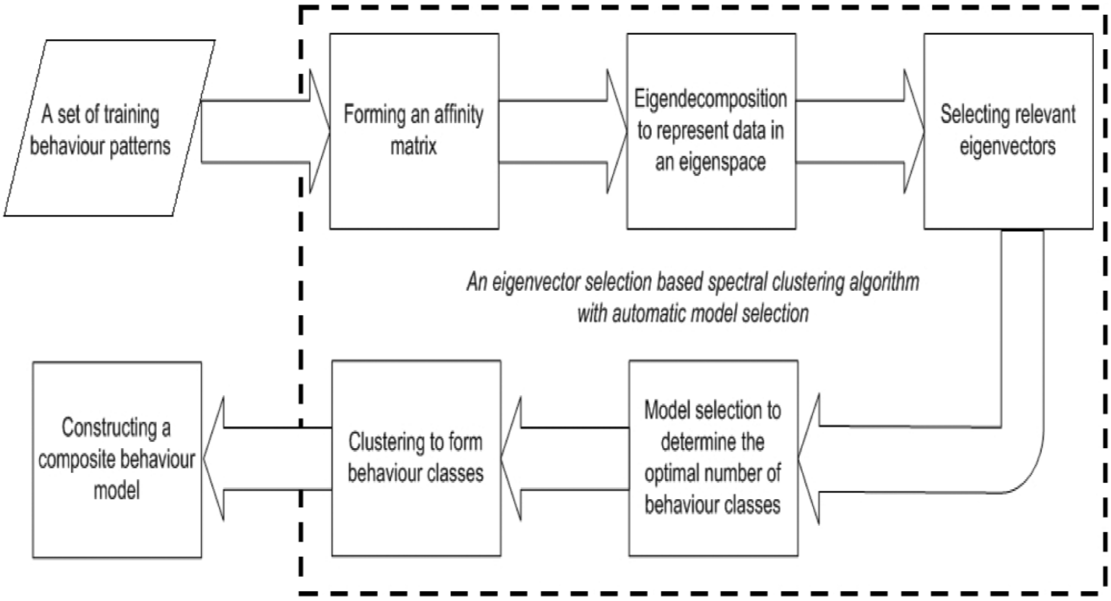

A set of training behaviour patterns
Forming an affinity matrix
Eigendecomposition to represent data in an eigenspace
Selecting relevant eigenvectors
An eigenvector selection based spectral clustering algorithm with automatic model selection
Constructing a composite behaviour model
Clustering to form behaviour classes
Model selection to determine the optimal number of behaviour classes

Figure 3.1 - Behavior Profiling Block Diagram

CHAPTER 4 ANOMALY DETECTION

4.1 INTRODUCTION: DETECTION OF ANOMALY

"Detection of Anomaly" is basically a method used to recognize unusual patterns which are fundamentally different from expected behavior, which are called "Outliers".This can be used in various business fields, likeintrusion detection (which is identifying the unusual patterns in network traffic that can signal a hack) to health monitoring system and from detecting fraud in credit/debit card's transactions to detection of fault in working environments.

4.2 NEED FOR ANOMALY DETECTION?

Machine learning consists of four basic classes of its applications: predicting the next value, classification, discovering the structure and anomaly detection. Amongst these, "Anomaly detection" identifies the points in our data that does not conform with rest of the data. This has an extensive range of the areas, where it finds its applications, for example, detection of frauds, surveillance, diagnosis, data cleanup, and predictive maintenance.

In spite of the fact that the concept of Anomaly detection has already been studied in detail theoretically, the areas of applications of "Anomaly detection" have been restricted to special areas like financial institutions, banks, medical diagnosis and auditingetc. However, after the advancement of the IOT, "Anomaly detection" would be liable to play a big role in IOT used issues, for example, predicting and monitoring maintenance.

This chapter deals with what is "Anomaly detection", what are the different techniques for "Anomaly detection", talks about the main idea behind these techniques, and ends up with a argument on the most proficient method to make utilization of those outcomes.

4.3IS IT NOT JUST CLASSIFICATION?

The answer to the above question will be yes if we satisfy the following three conditions.

1. We should have marked trained data.
2. The abnormal and the normal classes are properly balanced. (which should not be less than 1:5)
3. If there is no auto correlated data. (That existing data point does not rely on previous data points. This frequently breaks in time series data).

In the event that all of above is valid, we need not bother with the need of any "Anomaly detection technique and we can utilize a method/algorithm like Random Forests or Support vector machines (SVM).

Nevertheless, most of the times it is quite difficult to find the trained data, and even when one becomes successful in finding them, a good number of anomalies are 1:1000 to 1:10^6 occasions where classes are unbalanced. In addition, nearly all of the data, for example, the IOT use cases data, would be auto correlated.

An additional viewpoint is that the false positives are a noteworthy concern, as we will examine them under execution of decisions. Subsequently, the accuracy (given model anticipated an anomaly, how much likely it is for it to be true) and recollection (how much total anomalies the model is able to catch) trade-offs are not quite the same as normal classification cases.

4.4 WHAT ARE ANOMALIES?

Anomalies or outliers can be broadly categorized in the following three categories as:

1) **Point Anomalies**: If there exists a situation that a single data illustration can be taken to beanomalous in terms of the remaining data. (e.g. buy with vast exchange esteem). Business use case: Detecting the credit and/or debit card fraud based on the "transactions".

2) **Contextual Anomalies**: If a given data illustration is anomalous in a particular circumstance, however not-otherwise (anomaly if it happens at given time or given area. e.g. vast spike in the midnight). Such type of anomaly is very frequent

in time series figures. Business use case: Spending Rs.15000 on food and eateries daily during the holiday season is normal, but may not be normal otherwise and in other circumstances.

3) **Collective Anomalies**: In the event that a group of inter-related data instances is anomalous as compared to the complete data-set, but not the individual instance. Business use case: If somebody is attempting to replicate the data form a distant machine to a local host unexpectedly, an anomaly would be spotted in the form of a potential cyber attack. There are two variations namely,

   a) Events of an order are unexpected. (e.g. breaking of rhythm in an ECG)
   b) Value sets and combinations that are unexpected and unordered. (e.g. purchasing a lot of costly things)

"Anomaly detection" is analogous to novelty detection and removal of noise, but not entirely the same. **Novelty Detection** involves the identification of an overlooked pattern in absolutely new observations, which are not included in training data — like an immediate interest for some new video channel on YouTube during the time of Christmas, as example. The process of **Noise removal** (NR) consists of immunizing study by the presence of unprecedented observations; in simple words, removal of noise off a signal which is otherwise meaningful.

In the following segment, we will talk about exhaustively that how to deal with the collective and point anomalies. "Contextual Anomalies" are figured out by concentrating on data segments (e.g. spatial area, customer segment, graphs and sequences) and implementing collective anomaly methods inside each data segment separately.

4.5 TECHNIQUES FOR ANOMALY DETECTION

"Anomaly Detection" can be done through several methods depending upon the circumstances and the kind and nature of data. The segment below consists of the classification of some of these methods.

4.5.1 APPROACH OF STATIC RULES

This is probably the most basic and the best approach to begin with. The approach is to recognize a set of known anomalies and after that compose set of laws to recognize them. Rules identification can be done by either the use of pattern mining techniques or a domain expert or both.

Static rules are put to use with the widely accepted supposition that anomalies follow the 80:20 rules, where almost all theanomalous events are associated with only a few anomaly types. In the event that the supposition is true, we can recognize most of the anomalies by discovering those rules by which the anomalies are capable to be described.

The implementation of such rules can be done by utilizing one of the following three techniques.

1. In the event that these rules are basic and no further inference is required of any nature. We can use any programming language to code them.
2. In the event that the decisions require any inference, we can take help of an expert system like Drools.
3. In the event that the decisions have time related conditions, you can complete the task by using a *Complex event Processing System* (e.g. WSO2 CEP, esper).

However, static rule based systems have a tendency to be complicated and weak. Besides, recognizing these rules is primarily an unpredictable and subjective task. Hence, themachine learning based approach and the statistical approach, which consequently understand the general rules, are more preferable than static rules.

4.6 WHEN TRAINING DATA IS GIVEN

Anomalies are uncommon under general circumstances. Henceforth, even if the training data is given, it happens frequently that there could be a good number of anomalies that exist among a large number (around a million) of regular data points. The classification methods which are standard, for example, "SVM" or "Random Forest" more or less will identify all the data as normal on the grounds that doing that will give a huge precision score such as precision is 99.9 if anomalies are one out of thousand.

Most of the times, the class disproportion can be resolved by utilizing an ensemble built by re-sampling the data at number of times. The approach is, first make a new datasets by considering each and every anomalous points in data and then including a division of standard data points (for example as four times as anomalous data points). At that point a classifier is constructed for every set of data, utilizing "SVM" or "Random Forest", and these classifiers are joined utilizing ensemble learning. This method and strategy has functioned admirably and delivered great outcomes.

In the case that the data points are auto-correlated with one another, at that point the regular classifiers will not function admirably. We take care of such cases by means of the time series classification procedures or "Recurrent Neural Networks".

4.7 WHEN TRAINING DATA IS NOT GIVEN

In the case, that the training data is not given, it is still practical to perform Anomaly Detection, utilizing semi-supervised and unsupervised learning. Nonetheless, in the wake of developing the model, we will be clue less as to how perfectly it is performing, because you don't have anything to check against it. Henceforth, the results of these techniques should be first tested in the field prior to setting them in the critical course.

4.7.1 NO TRAINING DATA: POINT ANOMALIES

There is just one field in data set in the case of "Point Anomalies". We utilize the concept of percentiles to recognize "Point Anomalies" having histograms and numeric data to identify Detecting Point Anomalies in categorical data. In an event if, we find uncommon field values or data ranges from the data and envisage them as anomalies in the case that it repeats again and again. For instance, if 98.50 percentile of my transaction value is Rs.9500, then we can figure out that any higher value transaction than the normal value as a possible anomaly. While developing models, frequently we utilize the concept of moving averages rather than point values whenever we can because they have more stability towards noise.

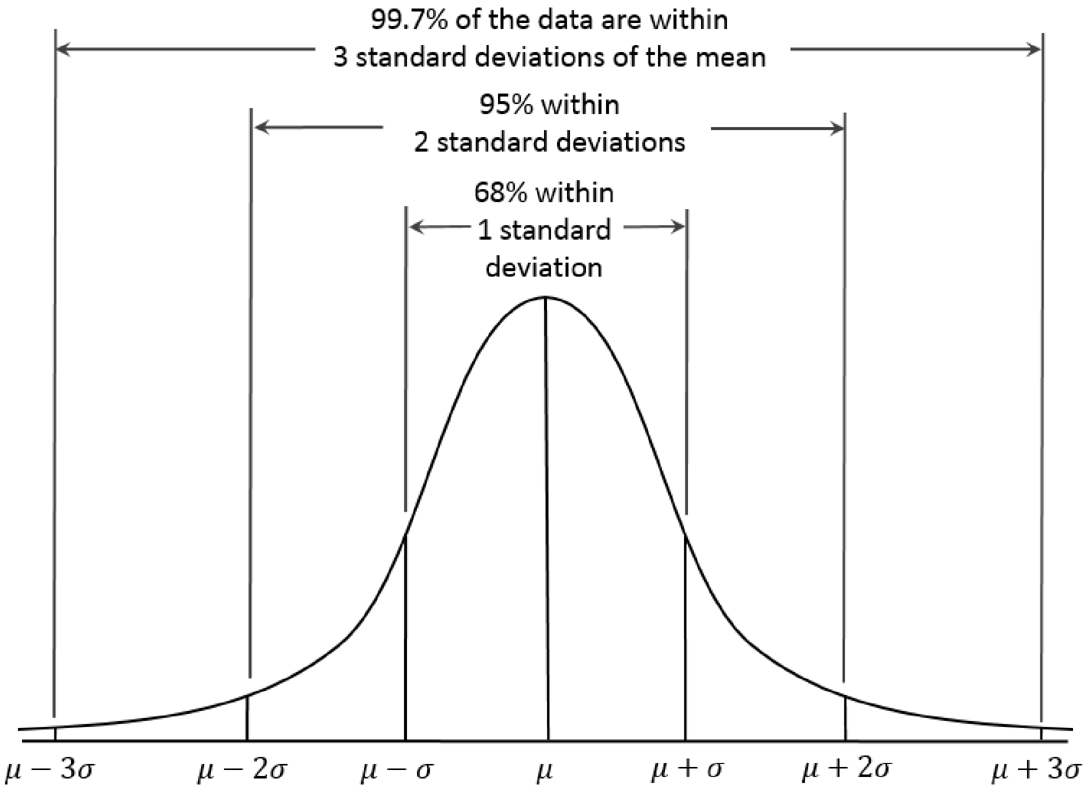


Figure 4.1 – Deviations in point anomalies

4.7.2 NO TRAINING DATA: UNIVARIATE COLLECTIVE OUTLIERS

Time series data is one of the best examples for collective anomalies in a univariate set of data. For such a situation, anomalies occur because of the fact that the values come in an order that is not expected. For instance, the 3rd heart beat may not be normal not on the grounds that the values are not in range, but rather occur in an incorrect order.

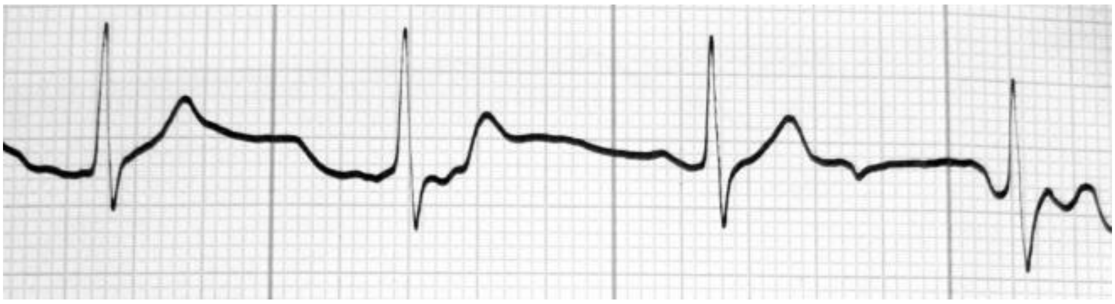


Figure 4.2 – Studying heartbeats for Anomaly

There are three different ways to deal with such use cases.

Solution 1: Create a predictor and search for anomalies with the help of residues: This depends on the heuristic that the values that are not clarified by the model will be termed as anomalies. Henceforth we can create a model that tells the following value, and after that apply percentiles on the errors (predicted value – actual value) as explained earlier. The model can be created by means of the concept of time series models, regression or Recurrent Neural Networks.

Solution 2: Markov chains & Hidden Markov chains can detect the possibility of the happening of a series of events. This idea creates a Markov chain for the underlined process, and when a series of events happens, we can utilize theMarkov chain to calculate the possibility of that series happening, and utilize that to spot any uncommon series.

For instance, let's take into account the credit / debit card transactions. To built the model for the credit / debit card transactions by means of Markov chains, let's symbolize every credit / debit card transaction, by means of 2 different values: the transaction value (L H) and the time since the very last transaction (L H). Since theMarkov chain's states must be limited in number, we will pick 2 values Low (L), High (H) to symbolize the variable values. Now the Markov chains will be represented by the states LL, LH, HL, HH and every transaction will be a transition from one state to the other state. We can create theMarkov chain by making use of the historical data and then using the created Markov chain we can calculate the sequence probabilities. After that, we can calculate the possibility of the happening of any new series and then highlight the uncommon series as anomalies.

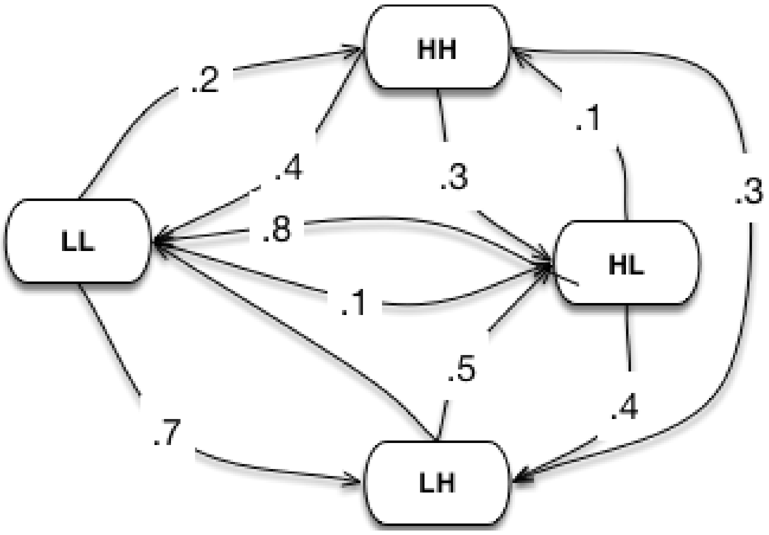


Figure 4.3 - Credit card transaction explained with Markov models

4.7.3 NO TRAINING DATA GIVEN: MULTIVARIATE COLLECTIVE OUTLIERS (UNORDERED)

In this case the data has various readings however, it doesn't have any order. For instance: figures and facts gathered from many individuals are so multi-variate yet the data set is not ordered. For instance: a moderate heartbeats and higher temperatures may be an anomaly despite the fact that both heartbeat and temperature, when considered alone, are in their typical range.

Approach 1: Clustering – In this approach the data is arranged in a cluster. Then the data found to be normal will belong to the formed clusters and the anomalies will either belong to the small clusters or will not be a part of any of the formed clusters.

At that point, to identify the anomalies, we will arrange our data in a clusters, and then compute the centroids and the density of each found cluster. After one gets another data point, the distance from the large known clusters to the new data point is computed and if the distance is too high then it is decided to be an anomaly.

Moreover, we should enhance this approach, initially by way of manual investigation of ranges of every cluster and then marking every cluster as normal or anomalous and utilize that during the process of checking anomaly for each data point.

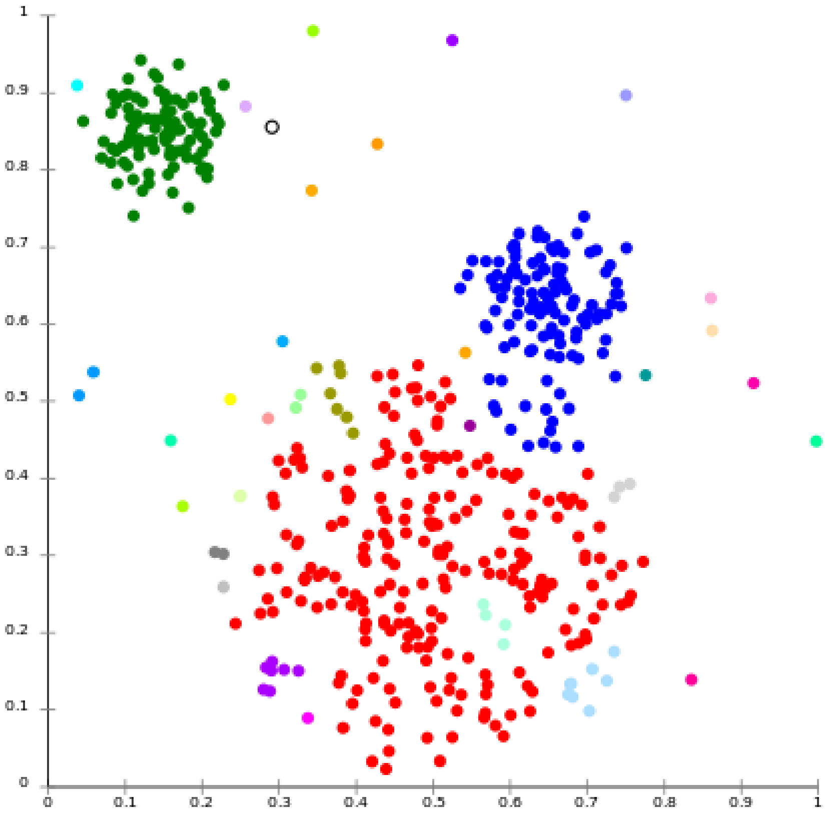


Figure 4.4 – Clustering Approach

Approach 2: Nearest neighbor techniques – In this approach it is assumed that the newly detected anomalies are nearer to the known anomalies. This approach can also be realized by utilizing the k-anomalies distance or utilizing the relative density of different anomalies that are closer to the fresh data points. While working out the above, along with the numerical data, we should break the space into many hypercubes, and with the definite data, the domain shall be broken down into bins by utilizing the histograms.

4.7.4 NO TRAINING DATA GIVEN: MULTIVARIATE COLLECTIVE OUTLIERS (ORDERED)

When no trained data is given then this class is considered as most general in ordering and in addition value combination. For instance, consider a sequence of important observations taken from the same patient. Some of the readings might be typical in combination, however anomalous because combinations occur in the incorrect order. For instance, if we are given the temperature, heart beat frequency, and blood pressure, each and every reading independent from anyone else might be normal, but not normal in the event that it oscillates too quick in a brief time frame.

Combine Markov Chains and Clustering – In this technique we combine clustering and Markov chains by first clustering the given data, & afterwards utilizing theclusters in the form of states in a Markov chain and creating a Markov chain. Markov chains will capture the order while Clustering will help in catching the common value combinations.

4.8 OTHER TECHNIQUES

There are various techniques in addition to those mentioned above that have been tested out, and following are some of those:

Information Theory: The fundamental thought of information theory is, the anomalies have a lot of information content because of inconsistencies, and the technique attempts to discover a division of data points which has the most number of irregularities.

Dimension Reduction: The fundamental thought in this technique is that after we apply this technique, the normal data is effortlessly expressed to be a mixture of dimensions while the anomalies have a tendency to build up complicated combinations.

Graph Analysis: Few procedures can have collaboration amongst various players. For instance, cash exchanges will build up a adjunct graph among the members. Flow analysis of these graphs may provide some anomalies. On some different use cases, for example, share markets, insurance, frauds, commercial payments and so forth, the similarities amongst transactions of players, may suggest someanomalous behavior.

4.9 COMPARING THESE MODELS AND RESULT ACTIONS

With the detection of anomaly, this is very normal to imagine that the fundamental objective is to identify each and every anomaly. However, this is a very frequent misconception.

There is a book named, "Statistics Done Wrong", that has an awesome illustration exhibiting the issue. How about we consider that there are 1000 patients and 9 of them suffer from the problem of breast cancer. There exists a test that detects the disease, which will catch approx. 85% of the patients, who have cancer (true positives). However, it also gives a positive response for approx. 10% of perfectly healthy patients (false positives).

This can be illustrated with the help of the following confusion matrix.

Table 4.1 Confusion Matrix

	Healthy	Unhealthy
Predicted Healthy	9190	1
Predicted Unhealthy	901	9

In such a circumstance, when the result tells that somebody has the cancer, in reality he doesn't have the cancer at 99% of the time. So this test is completely futile. If we go on to detect every anomaly, we will build up the same situation.

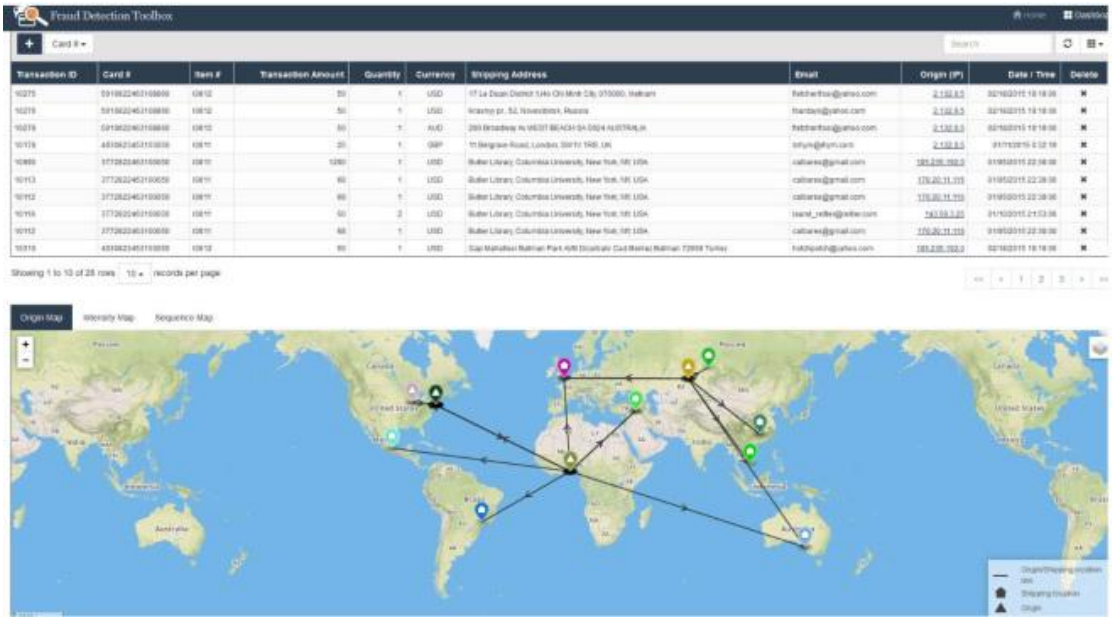
If you overlook the above issue, it can be harmful in numerous ways.

1. Lessen the confidence on the system – If individuals lose the trust, it will take a large number of threats and red tape to make them believe in it
2. Might do harm than good – in this illustration, emotional trauma and the tests that are not necessary may outweigh any advantages.
3. Might not be fair (e.g. scrutiny, arrest)

Consequently, we should attempt to discover an equilibrium where we attempt to identify what can be done while keeping the model precision within the satisfactory limits.

The other part of the same issue is, the models are just a recommendation for examination, however they are not a proof for inculpating somebody. It is one other loop of Correlation versus Causality. Hence, the results of this paradigm should not be utilized as proof and the examiner should discover autonomous proof of the issue (e.g. Credit / Debit Card Fraud).

Because of these reasons, it is foremost that examiner ought to have the capacity to check the anomaly inside the framework to verify it and furthermore to find the proof that something is not right. For instance, in WSO2 *Fraud Detection Solution*, analysts could tap on the warning of fraud and see the data point inside the framework of other data as demonstrated as follows:



Transaction ID	Card #	Item #	Transaction Amount	Quantity	Currency	Shipping Address	Email	Origin (IP)	Date / Time	Delete
[illegible]	[illegible]	[illegible]	50	1	USD	17 Le Duan District 1,Ho Chi Minh City, 070000, Vietnam	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	50	1	USD	Krasnyy pr. 52, Novosibirsk, Russia	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	AUD	[illegible] WEST BEACH SA 5024 AUSTRALIA	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	GBP	[illegible] Road, London, SW1V 1RB, UK	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	USD	Butler Library, Columbia University, New York, NY, USA	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	USD	Butler Library, Columbia University, New York, NY, USA	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	USD	Butler Library, Columbia University, New York, NY, USA	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	2	USD	Butler Library, Columbia University, New York, NY, USA	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	USD	Butler Library, Columbia University, New York, NY, USA	[illegible]	[illegible]	[illegible]	✖
[illegible]	[illegible]	[illegible]	[illegible]	1	USD	[illegible] Batman Park AVM [illegible] Batman [illegible] Turkey	[illegible]	[illegible]	[illegible]	✖

Showing 1 to 10 of 28 rows 10 records per page

« ‹ 1 2 3 › »

Figure 4.5 – Fraud detection toolbox by WSO2

Besides, having methods like unsupervised technologies and static rules, it is very hard to anticipate as to how many warning, this methods may lead to. For instance, it is not helpful for a group of 10 individuals to receive thousands or lakhs of alerts. We can deal with the issue by tracking the percentile on the anomaly score and just paying attention to the top most 1% of the time. In the event that the considered data is huge, we can utilize t-digest: as a percentile approximation technique.

Finally, we should focus on what analysts did with the warnings and enhance their experience. For instance, providing with the facility of auto-silencing of repeated warnings and alert digests etc. are the approaches that can give much control to the analysts.

4.10 DATASETS AND TOOLS

"Anomaly Detection" is generally performed with the traditional code& proprietary solutions. Henceforth, its use has been constrained to only a couple of high-value cases. In case we are searching for a solution which is open source, Some of the Options are given below:-

1. WSO2 has been working on the Fraud Detection tool based on top of WSO2 Data Analytics Platform. It comes free under Apache License.
2. Kale and Thyme by etasy provide support for anomaly detection based on time series.

Finally, only a couple of datasets, which are present in the public domain which can be utilized to test the anomaly detection problems, are there. It restricts the advancement of these methods and techniques.
Finally, the rundown architecture of anomaly detection is demonstrated below:

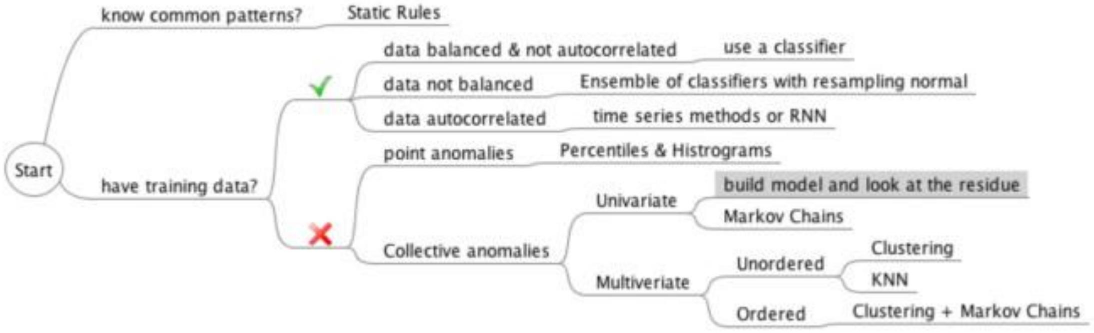


Figure 4.6 – Run down architecture of Anomaly Detection

CHAPTER 5 FINITE ELEMENT METHOD & FEM TOOLBOX

5.1 FINITE ELEMENT METHOD

5.1.1 INTRODUCTION

There exists a set of problems that are governed by elliptical partial differential equations. Since partial differential equations are involved in these problems and have specific boundary conditions, therefore these problems are known as boundary value problems. To find an approximate solution for these type of problems we have a numerical technique called as Finite Element Method(FEM). What the Finite element Method does is that it converts the difficult to solve set of elliptic partial differential equations into a set of easy to solve algebraic equations. However, the set of problems which consist of a hyperbolic or a parabolic differential equation and besides the boundary conditions, also consist of the initial conditions are called initial value problems. The Finite Element Method cannot solve such problems completely. Time is one of the independent variables in the hyperbolic or the parabolic differential equations. Finite Difference Method, abbreviated as FDM is another numerical technique, which is required to convert the time or temporal derivatives into simpler algebraic expressions. Thus, one needs both, the Finite Element Method as well as the Finite Difference Method to solve an initial value problem. The Finite Element Method converts the spatial derivatives into algebraic expressions whereas the Finite Difference Method converts the temporal derivatives into algebraic expressions.

5.1.2 HISTORICAL BACKGROUND

The words "finite element method" were first used by Clough in his paper in the Proceedings of 2ndASCe (American Society of Civil engineering) conference on electronic Computation in 1960. Clough extended the matrix method of structural analysis, used essentially for frame-like structures, to two-dimensional continuum domains by dividing the domain into triangular elements and obtaining the stiffness matrices of these elements from the strain energy expressions by assuming a linear variation for the displacements over the element. Clough called this method as the finite element method because the domain was divided into elements of finite size.

(An element of infinitesimal size is used when a physical statement of some balance law needs to be converted into a mathematical equation, usually a differential equation).

Argyris, around the same time, developed similar technique in Germany . But, the idea of dividing the domain into a number of finite elements for the purpose of structural analysis is older. It was first used by Courant in 1943 while solving the problem of the torsion of non-circular shafts. Courant used the integral form of the balance law, namely the expression for the total potential energy instead of the differential form (i.e., the equilibrium equation). He divided the shaft cross-section into triangular elements and assumed a linear variation for the primary variable (i.e., the stress function) over the domain. The unknown constants in the linear variation were obtained by minimizing the total potential energy expression. The Courant's technique is called as applied mathematician's version of FEM where as that of Clough and Argyris is called as engineer's version of FEM.

From 1960 to 1975, the FEM was developed in the following directions :

(1) FEM was extended from a static, small deformation, elastic problems to

- dynamic (i.e., vibration and transient) problems,
- small deformation fracture, contact and elastic -plastic problems,
- non-structural problems like fluid flow and heat transfer problems.

(2) In structural problems, the integral form of the balance law namely the total potential energy expression is used to develop the finite element equations. For solving non-structural problems like the fluid flow and heat transfer problems, the integral form of the balance law was developed using the weighted residual method.

(3) FEM packages like NASTRAN, ANSYS, and ABAQUS etc. were developed.

The large deformation (i.e., geometrically non-linear) structural problems, where the domain changes significantly, were solved by FEM only around 1976 using the updated Lagrangian formulation. This technique was soon extended to other problems containing geometric non-linearity :

- dynamic problems,
- fracture problems,
- contact problems,
- elastic-plastic (i.e., materially non-linear) problems.

Some new FEM packages for analyzing large deformation problems like LS-DYNA, DeFORM etc. were developed around this time. Further, the module for analyzing large deformation problems was incorporated in existing FEM packages like NASTRAN, ANSYS, ABAQUS etc

5.1.3 PROCEDURE OR STEPS FOR FEM

The Method of Finite Element involves a series of steps, which are listed below:

- First, we find out the integral form for the governing partial differential equation of the problem. There are two different techniques by which this can be done:
  - Variational Technique
  - Weighted Residual Technique.

  When we use calculus of variation in order to obtain the integral form of our differential equation then this technique is known as variational technique. Then we minimize that integral which leads to the solution of our problem.

  In weighted residual technique, When we obtain the integral form corresponding to the given differential equation as a weighted integral of our differential equation, then it is called as weighted residual method. Here the weight functions are already known and are also arbitrary, but they satisfy certain boundary conditions. In this case, there is also a requirement for continuity of the solution. To lessen this requirement, we modify our integral, and this task is usually done with the help of the divergence theorem. We set the integral form to zero so as to obtain the solution of our problem.

- In the second step of the Finite element Method, division is done. The domain of the problem is divided into various number of small parts. These parts are called elements. For the problems having only one dimension (1-D), the elements will be also obviously of one dimension only. Therefore they are just line segments, which have only length but have no shape. The elements will have both, the shape as well as the size for the problems of higher dimensions. For the problems with two dimensions (2-D), also known as axi-symmetric problems, quadrilateral, rectangles and triangles which have curved or straight boundaries are used as the elements. Shapes like tetrahedron, bricks and parallelepiped having curved surfaces are used for three dimensional problems. After forming elements into the domain of our problem, we obtain a mesh.

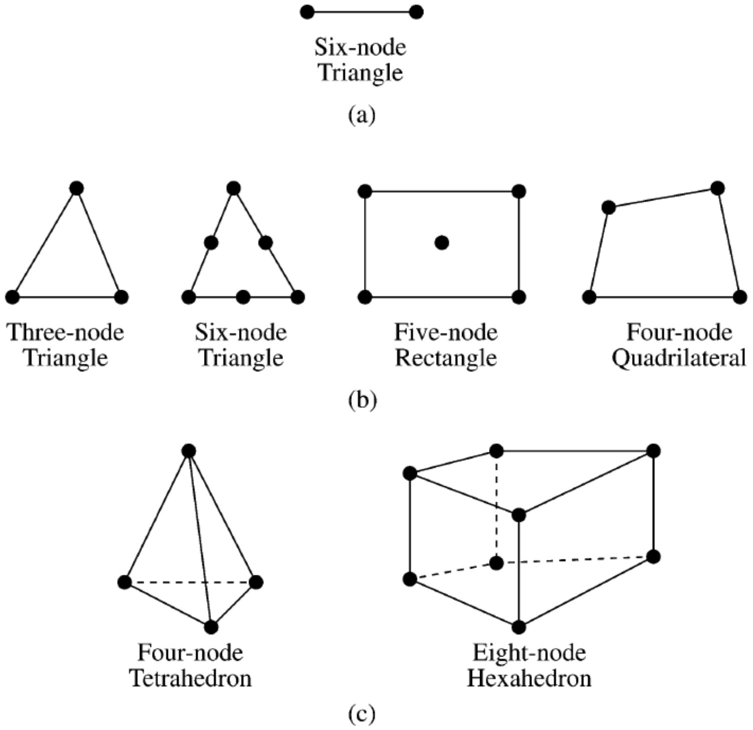


Figure 5.1 - Typical Finite Elements: (a) 1-D, (b) 2-D, (c) 3-D

- In the third step, we consider a single element, and with the help of interpolation/shape functions and the unknown values of the primary variable at pre-defined points; which are also called nodes; we choose an appropriate approximation for that element. for the shape functions, we often choose polynomials. There should be a minimum of 2 nodes for a one dimensional element and those too at the end points; and if we wish to add more nodes, then those nodes are added in between those end points. For two and three dimensional elements, we place the nodes at the vertices. For a triangular element, we need at least three nodes. For a rectangular or a tetrahedral element, we need at least four nodes. For three dimensional elements, we need minimum of 8 nodes. If we wish to add more nodes in the case of two and three dimensional elements, then those nodes are placed either at the edges or in the interior or both. At these nodes, the value of the above mentioned primary variable is known as the degree of freedom.

- We can obtain the exact solution in two situations:

  If the number of elements is finite, then the expression of the primary variable should have infinite number of terms, and

  If the expression of our primary variable does not have infinite number of terms, then the number of elements should be infinite.

- In either case, we want to obtain a set of algebraic equations which is infinite in number. For our convenience to solve this problem, we consider the primary variableexpression with finite number of terms and finite number of elements. Solving in this situation will give us an approximate solution. Nevertheless, we can improve the accuracy of our approximate solution by increasing the number of elements, or by increasing the number of terms in our approximation.

- In the final step, the approximation for the primary variable is put into the basic integral form. In the event that the integral form belongs to the variational sort, it is reduced to get the equations for the obscure nodal values of the primary variable. In the event that the integral form belongs to the weighted residual sort, it is set to zero to acquire the same set of equations. For each situation, the equations are acquired element savvy first (called as the element equations) and afterward they are collected over every one of the elements to get the equations for the entire space/domain (known as the global equations).

  - In this progression, the algebraic equations are altered to deal with the boundary conditions that are on the primary variable. The changed equations are fathomed to discover the nodal values of the primary variable.
  - In the last stride, the post-preparing of the solution is finished. That is, first the secondary variables of our problem are computed from the solution. At that point, the nodal values of the primary and auxiliary variables are utilized to build their graphical variation over the domain either as graphs (for 1-D) or 2-D/3-D shapes, all things considered.

5.1.4 ADVANTAGES OF FEM

- Merits of the finite element strategy over other numerical strategies are as per the following:
- The technique can be utilized for any irregularly shaped domain and a wide range of boundary conditions.
- Domains comprising of more than one material can be effortlessly examined.
- Accuracy of the solution can be enhanced either by appropriate refinement of the mesh or by picking approximation of higher degree polynomials.
- The arithmetical equations can be effectively produced and solved on a PC. Truth be told, a broadly useful code can be created for the investigation of an expansive class of problems.

5.2 FEM TOOLBOX

Matlab FEM Toolbox for Solid Mechanics is basically intended for intermediate-level users and it allows them to explore the power of FEM. This toolbox works in Windows, it can solve many linear problems in solid mechanics, and it provides the users with extensive visualization features. Sample input data are illustrated in a file with a built in structure, which is illustrated with the help of several examples that are included in the directory : \FEM\input.

5.2.1 BASIC FEATURES:

- Static, dynamic, modal analysis, and heat transfer for linear systems;
- Simple User Interface with input model visualization;
- Input file based on the data structures (nodes, elements, constraints, etc.);
- Result can be visualized in ways like mode shapes, deformations and response animations
- Maximum tested size of the model is : 307,686 DOFs, for static and modal analysis.

CHAPTER 6 PROPOSED METHODOLOGY

In this thesis, we will implement videos frame by frame with the help of FEM toolbox. Although FEM toolbox in MATLAB is used in the field of solid mechanics, here we will manipulate those applications to our use and check for anomalies in the videos, also we will compare and contrast different human actions.

At this time, the main question that is to be answered is that what will FEM toolbox (for solid mechanics) do to analyse an image, let alone a video. The answer to this question is that this toolbox will give us modal frequencies of respective video frames.

It is these frequencies which will help us work out or find out an anomaly in a video or compare and contrast between two human actions.

The data sets used for implementation of this method are KTH and WEIZMANN.

CHAPTER 7 IMPLEMENTATION & RESULTS

7.1 ACTIVITY RECOGNITION: COMPARE & CONTRAST

7.1.1 CLAPPING

For this action, we first take down all the nodal values of the concerned image. For instance, for the image given below, the nodal values are given in table (so and so).

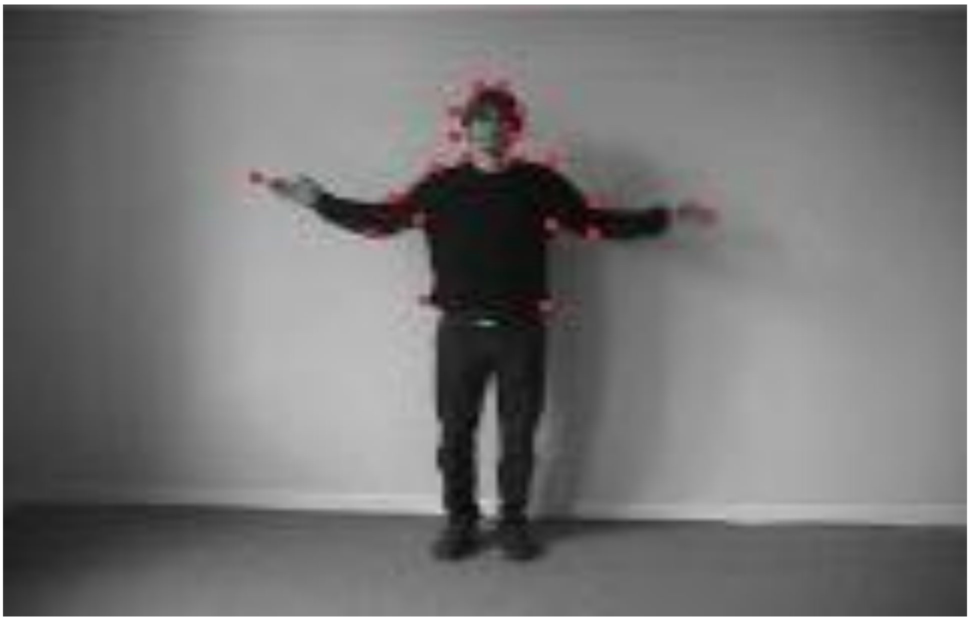


Figure 7.1 - Clapping Image Frame (Frame 6)

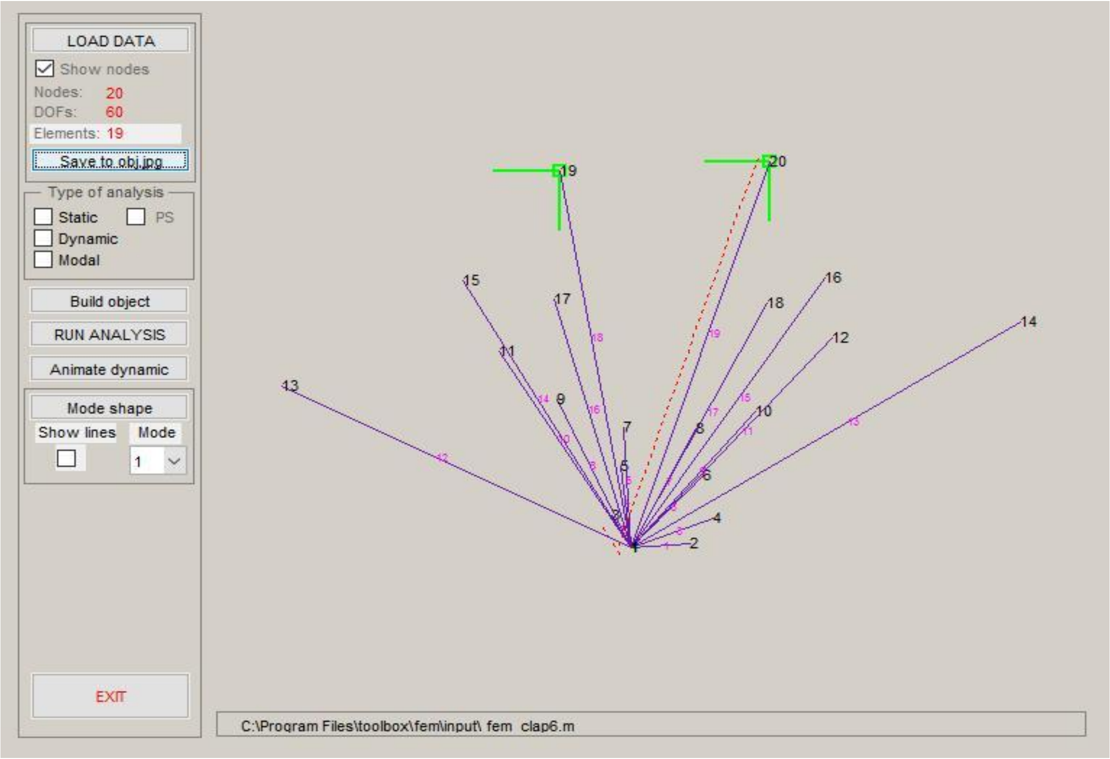
The red dots marked above are the nodes for one image frame of clapping. Their coordinates are as follows:

Table 7.1 Nodal Values of Frame 6

S.No	X	Y
1	78.38	17.05
2	84.35	17.55
3	76.39	20.54
4	86.59	20.29
5	77.38	26.01
6	85.60	25.02
7	77.63	30.25

8	84.85	30.00
9	70.91	33.23
10	90.83	31.99
11	65.18	38.46
12	98.54	39.96
13	43.28	34.73
14	117.47	41.70
15	61.45	46.18
16	97.80	46.43
17	70.66	44.19
18	92.07	43.69
19	71.16	58.13
20	92.32	59.13

Now, these coordinates are joined to form elements. In our case, we have joined our nodes keeping node number 1 as a reference and joining all the other nodes to that node number one.



☑ Show nodes
Nodes: 20
DOFs: 60
Elements: 19
Show lines □
Mode 1

C:\Program Files\toolbox\fem\input\ fem_clap6.m

Figure 7.2 - Nodes & Elements of Frame 6

After forming our elements, we load the corresponding function file into the FEM toolbox. After loading the image, we can see something like in figure above.

In order to get the modal frequency, we do the modal analysis; first for Mode 1(set as default when we launch the toolbox). The result for Mode 1 is as follows:

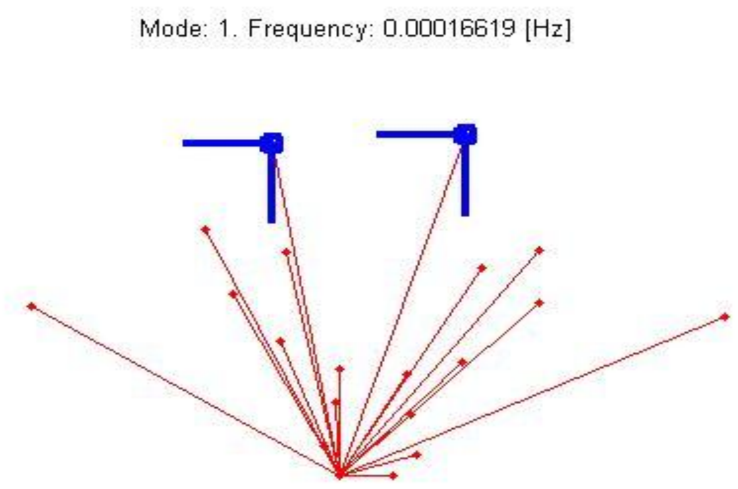


Figure 7.3 - Mode 1 frequency of Frame 6

Then for Mode 2 and 3 respectively

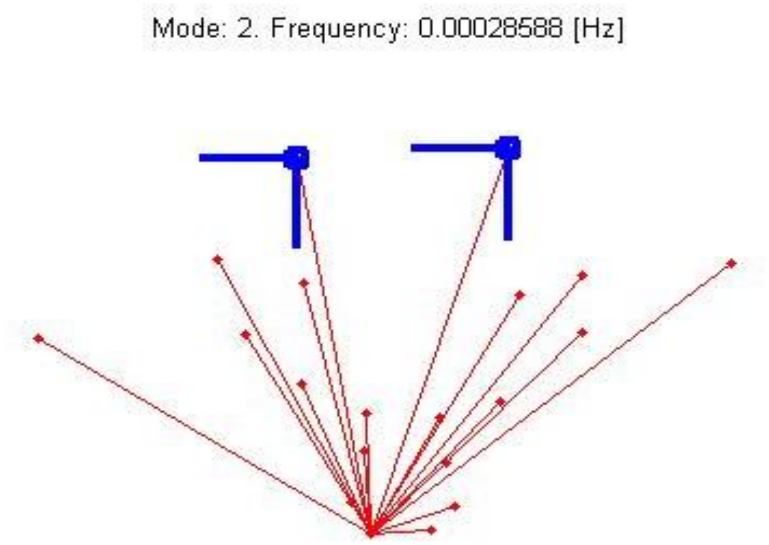


Figure 7.4 - Mode 2 frequency for Frame 6

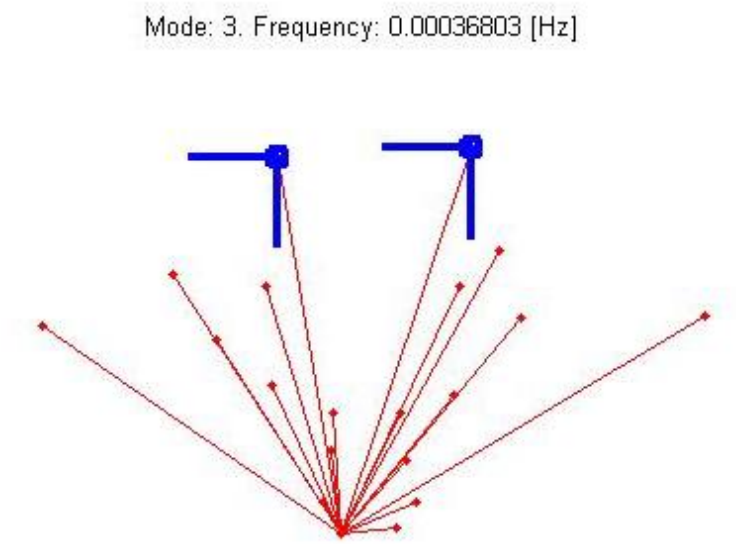


Figure 7.5 - Mode 3 frequency for Frame 6

Now , in a similar fashion, we calculate modal frequencies for all the 3 modes for all the 15 frames of our video which covers a single action of clapping (back and forth only once)

Table 7.2 Modal frequencies for all frames

CLAPPING (15 FRAMES) Frequency (*10^-5 Hz)			
FRAME NO.	MODE 1	MODE 2	MODE 3
1	23.158	40.554	48.898
2	20.699	40.937	43.987
3	19.724	39.432	45.306
4	17.812	35.147	38.604
5	16.676	29.901	36.436
6	16.619	28.588	36.803
7	16.599	28.485	37.276
8	16.76	28.995	37.109
9	16.625	28.718	37.086
10	16.702	28.931	37.694
11	17.419	33.399	39.072
12	17.543	36.826	39.106
13	18.85	39.631	41.901
14	20.024	39.209	41.325
15	21.811	39.104	45.264

Finally we plot graphs of frequency vs the frame number in order to check whether this particular action is periodic or not

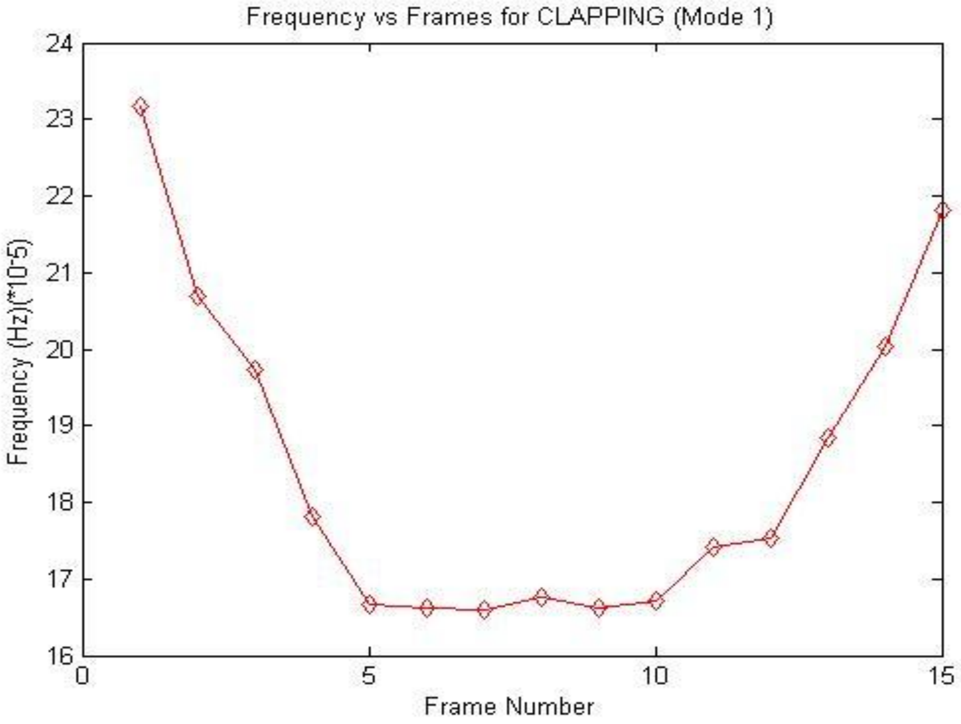


Figure 7.6 - Mode 1 Plot of frequencies vs Frames

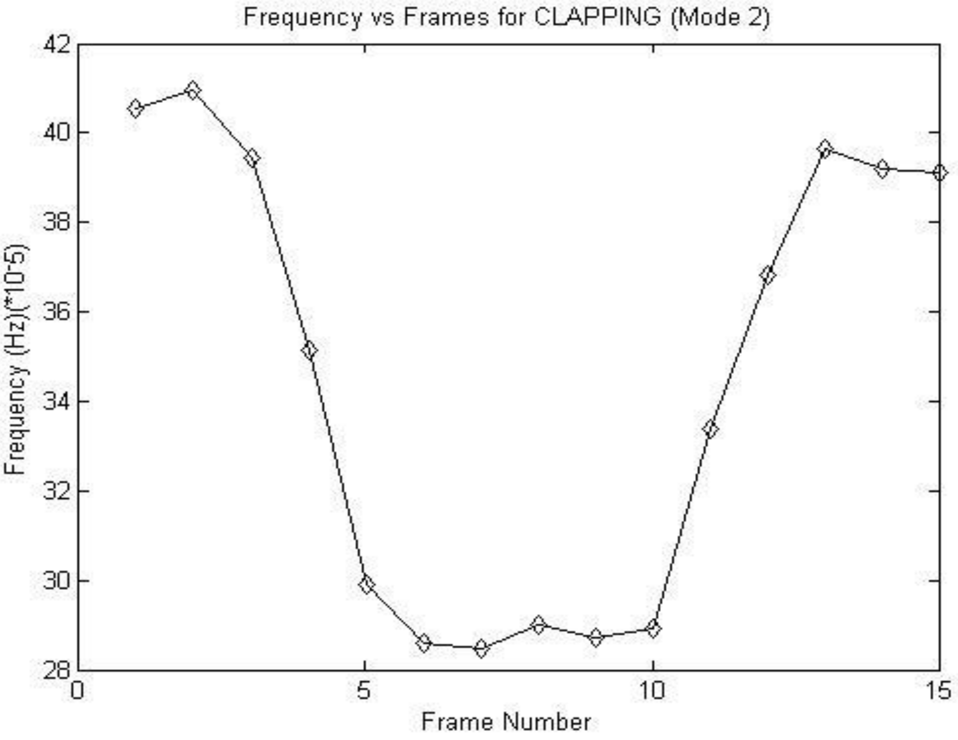


Figure 7.7 - Mode 2 plot of frequencies vs Frames

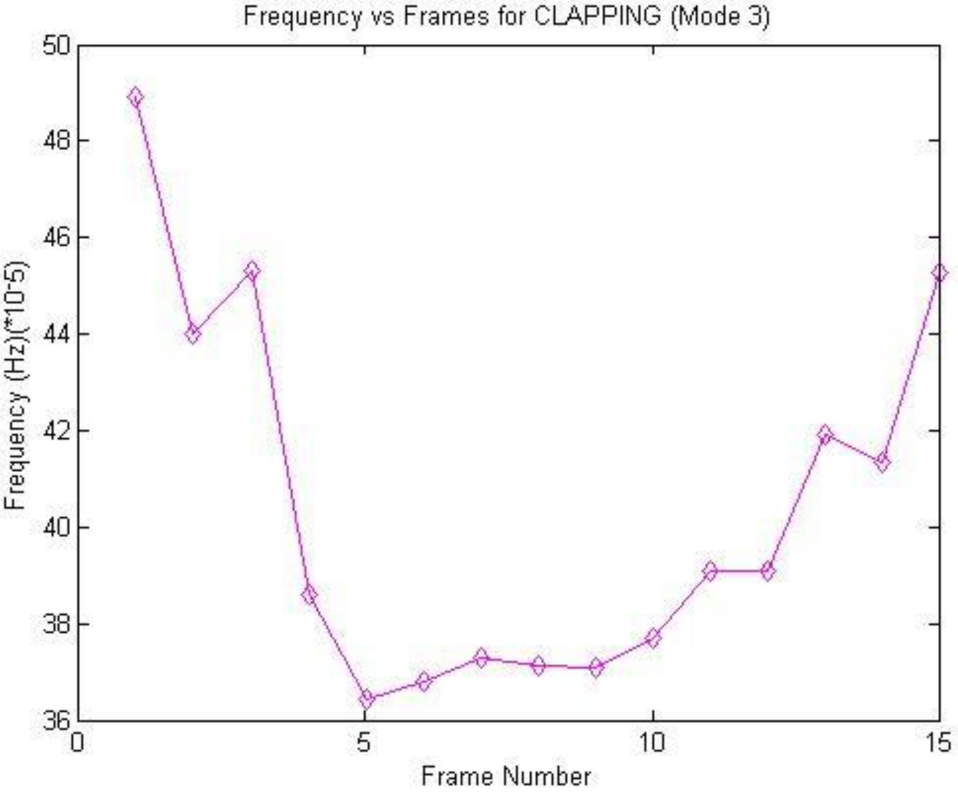


Figure 7.8 - Mode 3 plot of frequencies vs Frames

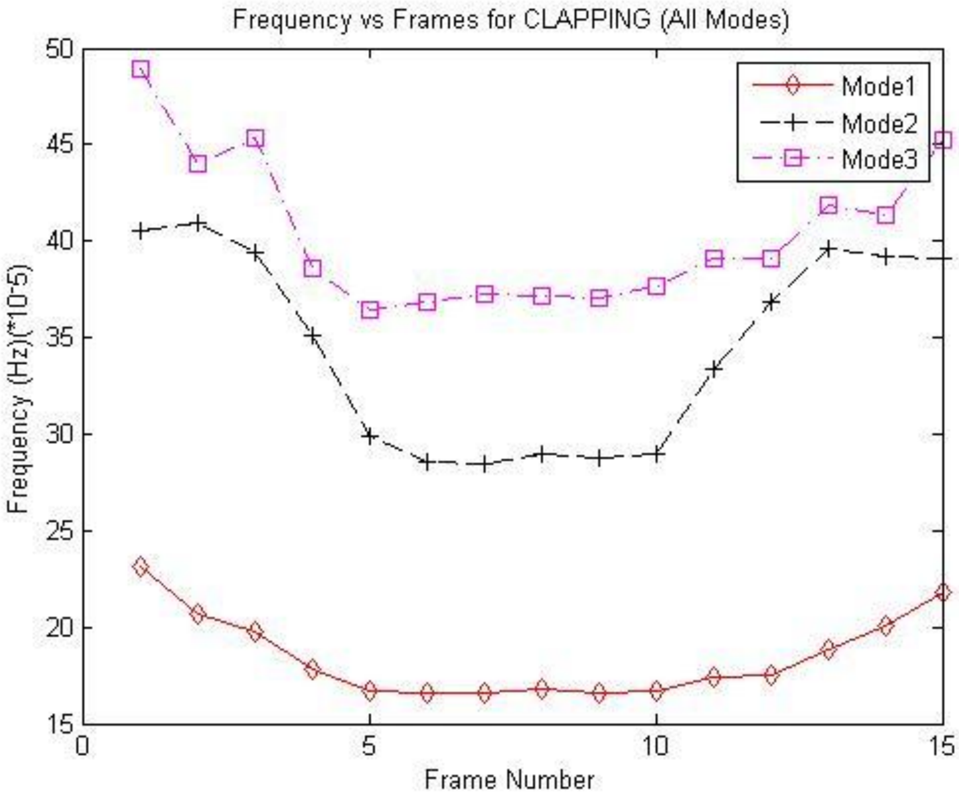


Figure 7.9 - All Modes

As we can see, the graphs of these modal frequencies show a periodic trend with slight distortions in Mode 2 and Mode 3 plots.

7.1.2 SKIPPING

In this action, we will analyse unidirectional movements in FEM. Following the same procedure, we have a sample frame given with nodes marked, below.

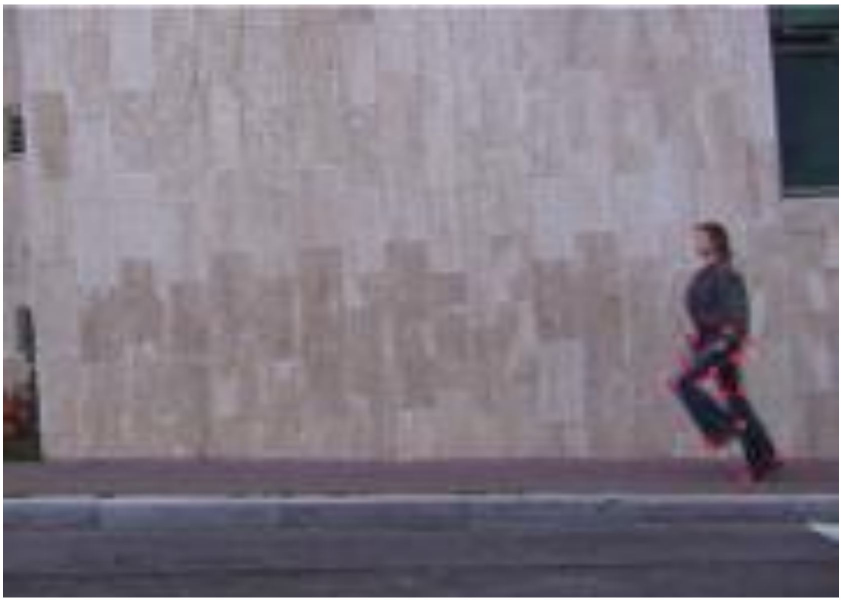


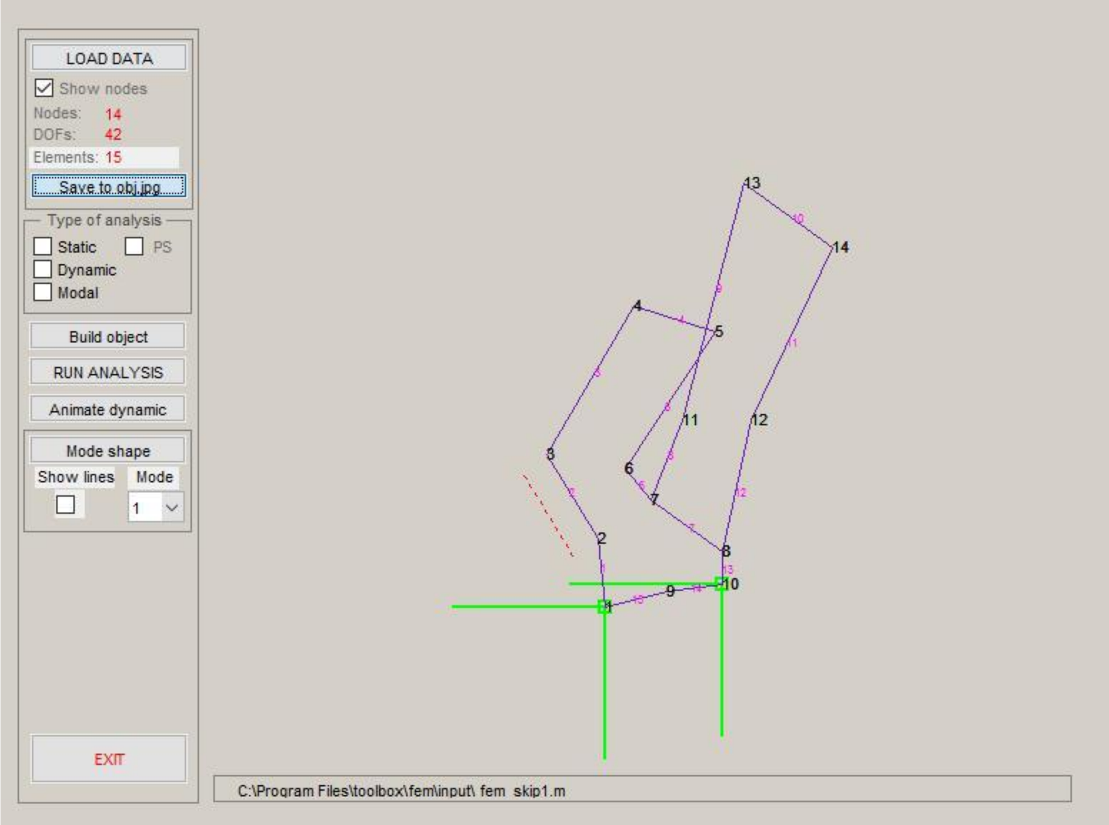
Figure 7.10 - Skipping Image Frame (Frame 5)

Corresponding nodal values are,

Table 7.3 Nodal values of Frame 5

S.No	X	Y
1	149.06	80.18
2	148.51	85.94
3	144.12	93.07
4	151.52	105.68
5	158.38	103.49
6	150.70	91.97
7	152.90	89.23
8	158.93	84.84
9	154.27	81.55
10	158.93	82.10
11	155.64	96.08
12	161.40	96.08
13	160.85	116.11
14	168.26	110.62

Now, these coordinates are joined to form elements. This time we have joined the nodes in order to form the shape of a person's legs while he/she skips. This is shown in the figure below.



☑ Show nodes
Nodes: 14
DOFs: 42
Elements: 15

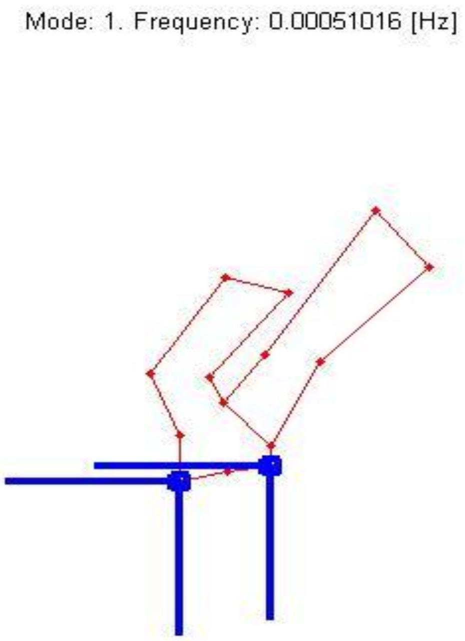
□ Static □ PS
□ Dynamic
□ Modal

Show lines □
Mode 1

C:\Program Files\toolbox\fem\input\ fem_skip1.m

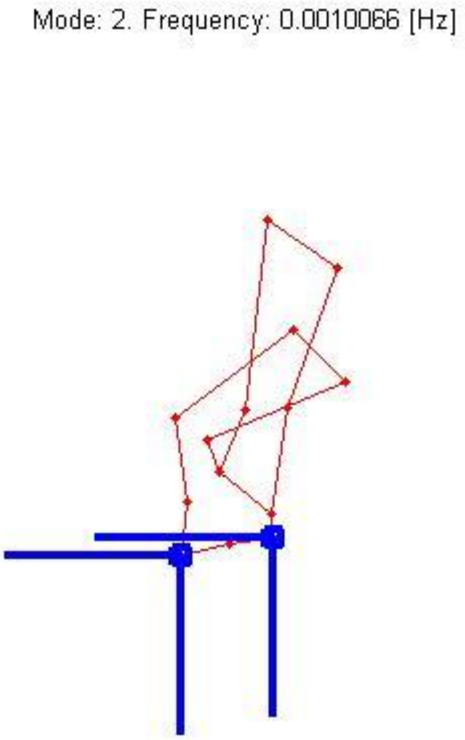
Figure 7.11 - Nodes & Elements of Frame 5

After Modal analysis for Mode 1,2 and 3, we get the following



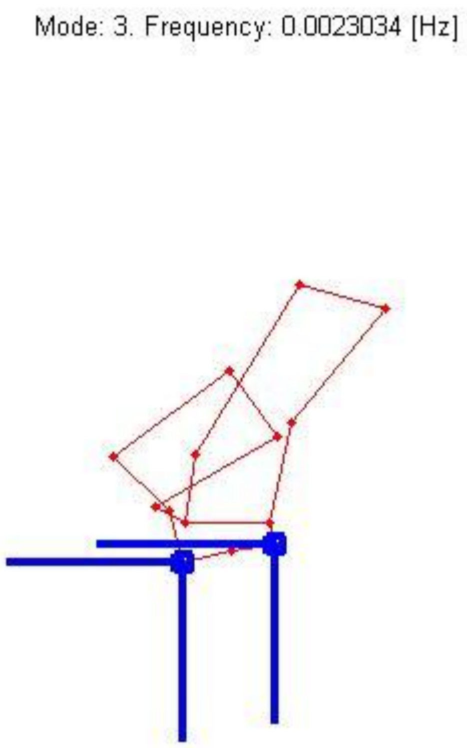
Mode: 1. Frequency: 0.00051016 [Hz]

Figure 7.12 - Mode 1 Frequency for Frame 5



Mode: 2. Frequency: 0.0010066 [Hz]

Figure 7.13 - Mode 2 Frequency for Frame 5



Mode: 3. Frequency: 0.0023034 [Hz]

Figure 7.14 - Mode 3 Frequency for Frame 5

Modal Frequencies of all the frames of one skipping action are given below.

Table 7.4 Modal Frequencies for all Frames

SKIPPING (11 FRAMES)			
FRAME NO.	MODE 1	MODE 2	MODE 3
1	48.855	109.650	22.479
2	46.255	100.980	21.819
3	49.114	81.494	20.998
4	55.049	103.170	23.617
5	51.016	100.660	23.034
6	50.674	102.840	19.533
7	41.283	79.699	20.280
8	46.325	91.804	18.658
9	41.812	94.517	19.480
10	37.086	78.842	19.006
11	41.175	79.795	20.722

Corresponding graphs are as follows.

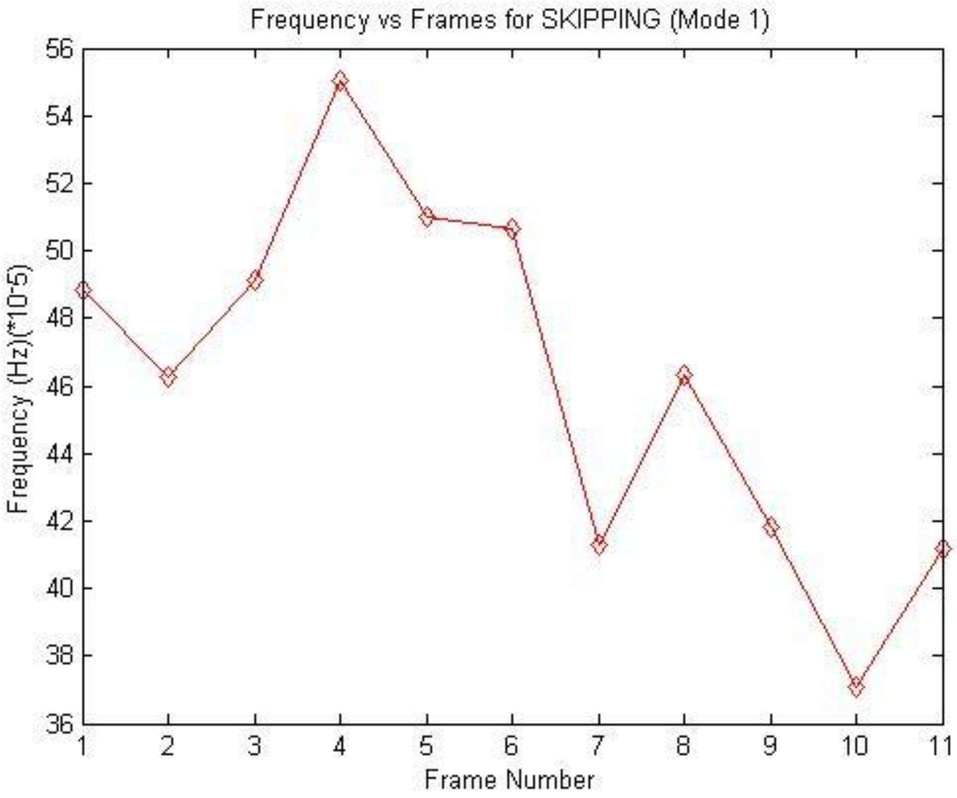


Figure 7.15 – Mode 1 plot of Frequencies vs Frames

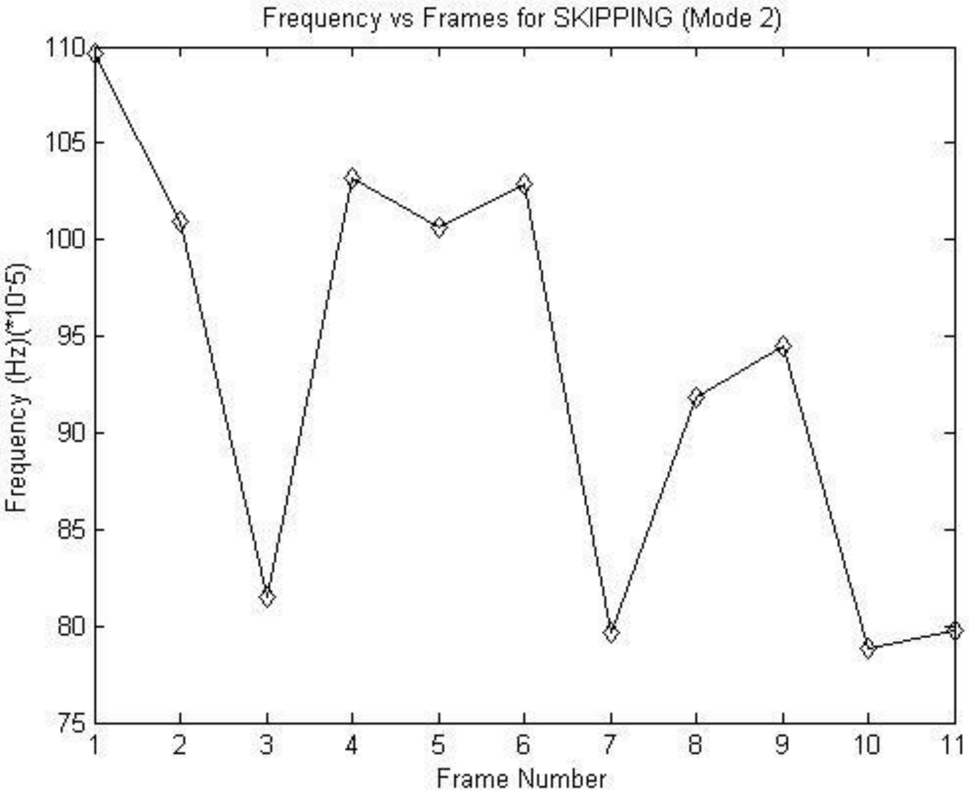


Figure 7.16 - Mode 2 plot of Frequencies vs Frames

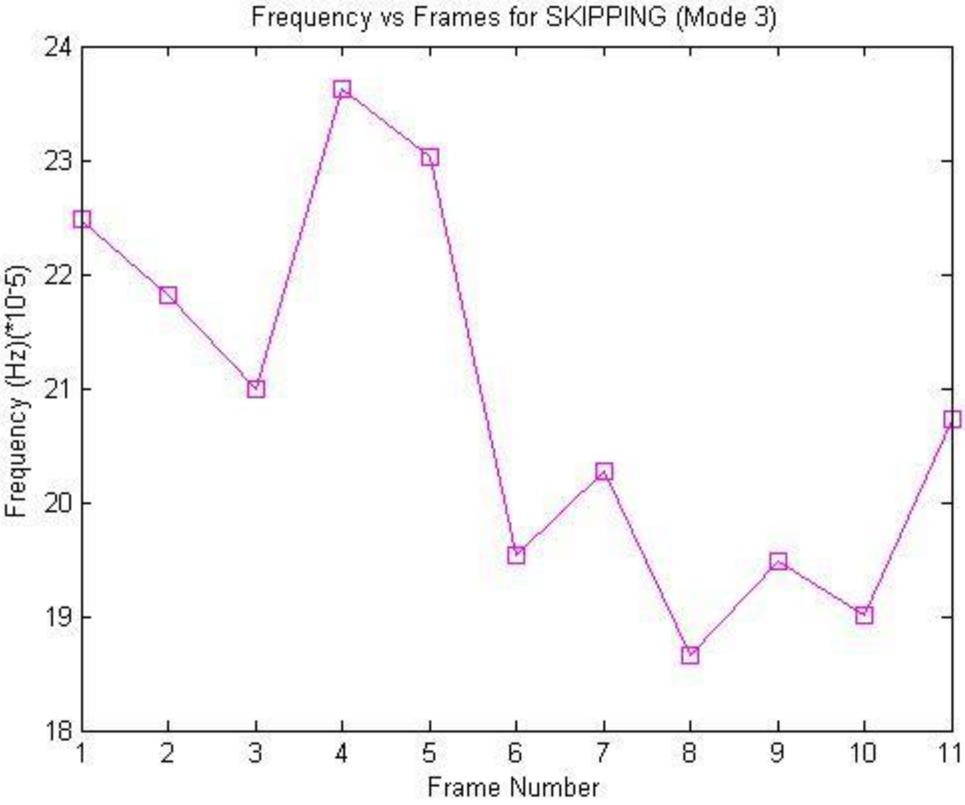


Figure 7.17- Mode 3 plot of Frequencies vs Frames

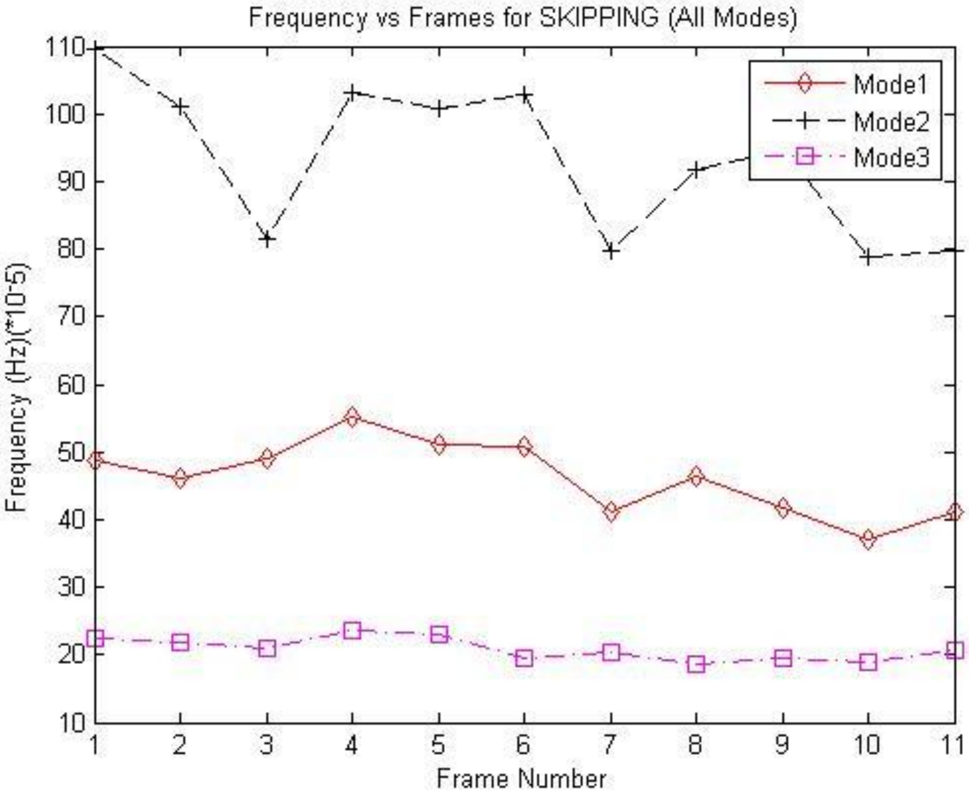


Figure 7.18- All Modes

From the above analysis and results, we can see that unidirectional actions like skipping or hopping are not periodic even if a person is repeating an exact action continuously.

Hence, we can now distinguish between clapping and skipping using the above method. Here the basis of differentiation is whether an action is periodic or not.

In the upcoming actions like jumping, hand waving with one or two hands, we will perform similar analysis and the results obtained will be same as that in clapping.

7.1.3JUMPING

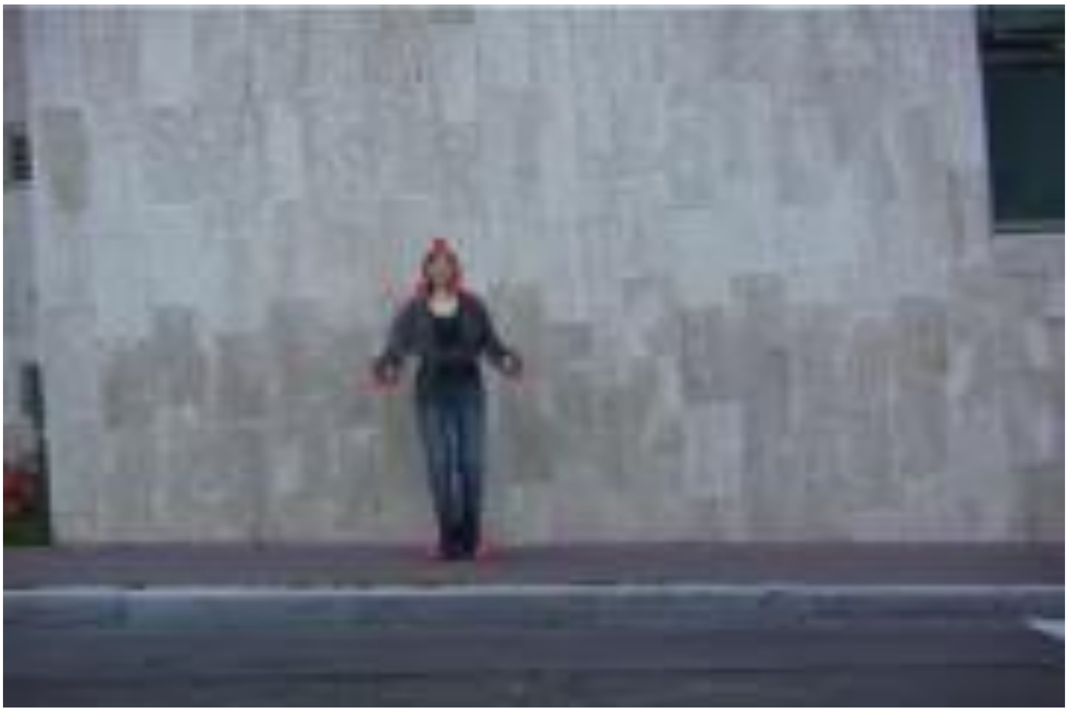


Figure 7.19 - Jumping Image Frame (Frame 2)

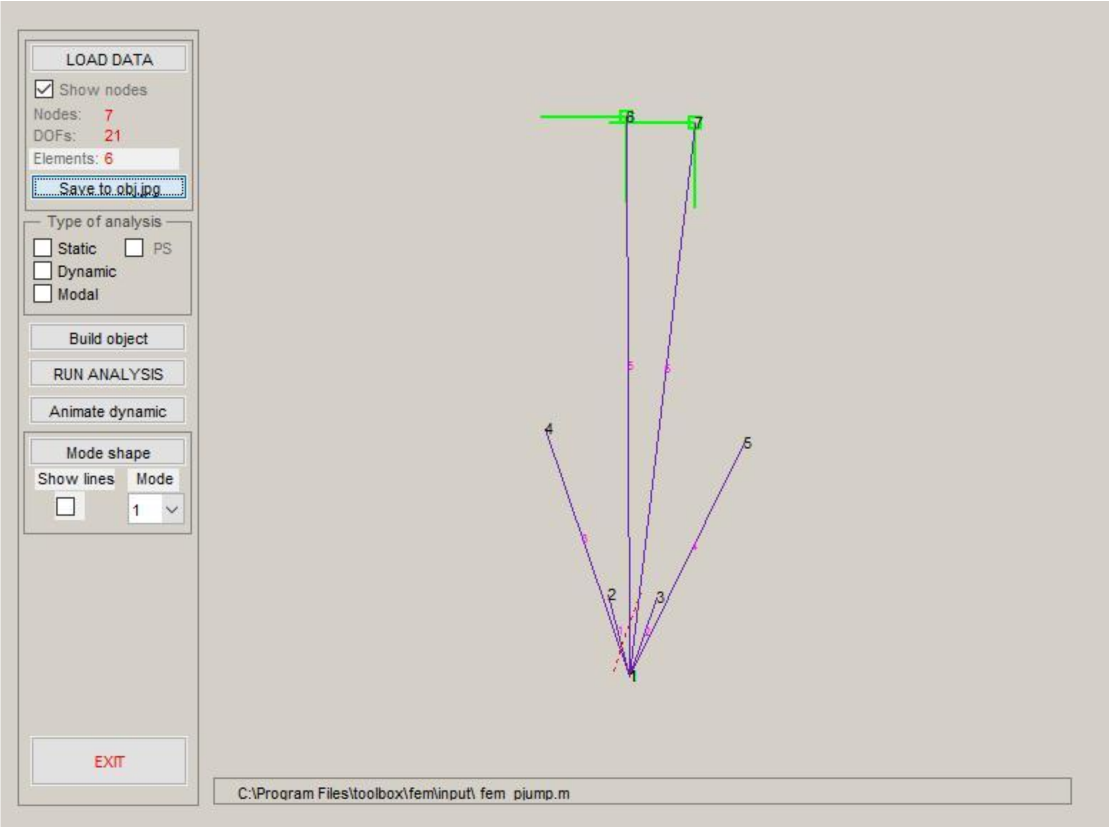
Following are the nodal values of the above frame.

Table 7.5 Nodal values of Frame 2

S.No	X	Y
1	74.50	49.88
2	72.03	59.11
3	77.57	58.80

4	64.96	77.88
5	87.42	76.34
6	74.19	113.26
7	81.88	112.65

After we input the above structure into FEM, we get the following image. Remember that elements are formed keeping one node as a reference and joining the rest with that reference node.



☑ Show nodes
Nodes: 7
DOFs: 21
Elements: 6

Type of analysis
□ Static □ PS
□ Dynamic
□ Modal

Show lines □
Mode 1

C:\Program Files\toolbox\fem\input\ fem_pjump.m

Figure 7.20 - Nodes & Elements of Frame 5

Performing Modal analysis, we get the following results

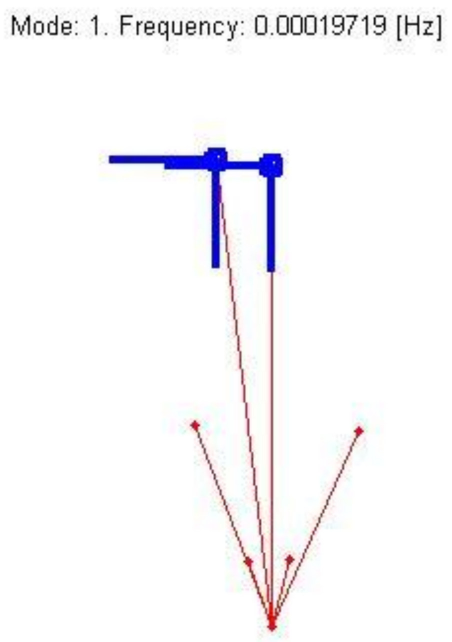


Figure 7.21 - Mode 1 Frequency for Frame 2

Similarly, for Mode 2 and Mode 3,

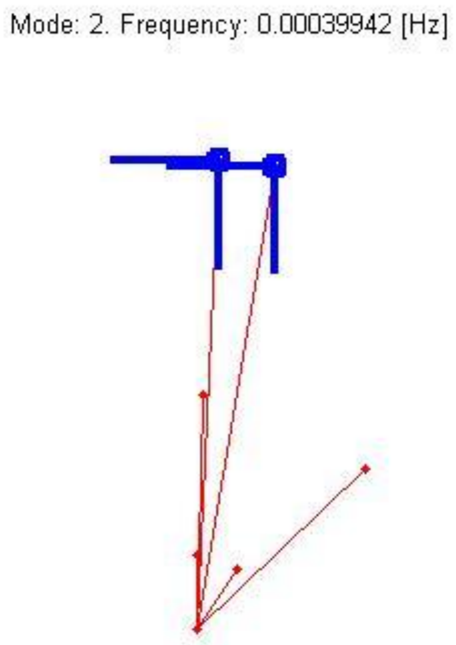
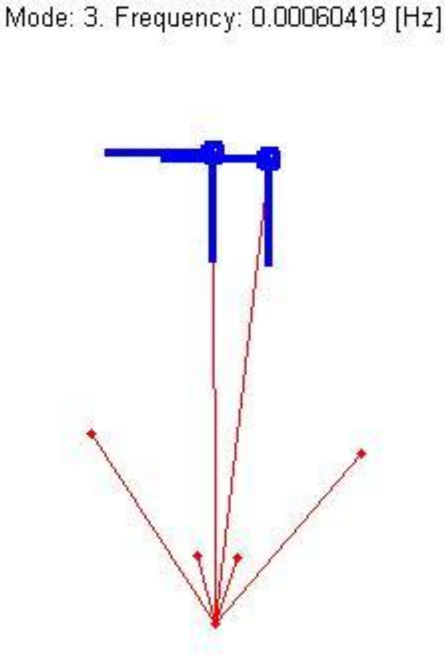


Figure 7.22 - Mode 2 Frequency for Frame 2



Mode: 3. Frequency: 0.00060419 [Hz]

Figure 7.23 - Mode 3 Frequency for Frame 2

After performing modal analysis on all the frames of jumping, we get

Table 7.6 Modal frequencies of all Frames

JUMPING (11 FRAMES)			
FRAME NO.	MODE 1	MODE 2	MODE 3
1	21.917	37.345	53.168
2	19.719	39.942	60.419
3	20.712	41.304	61.501
4	16.703	30.341	43.850
5	17.444	30.141	42.403
6	15.876	27.435	38.520
7	17.059	29.154	40.018
8	17.329	28.947	39.195
9	19.589	31.360	38.530
10	21.449	32.227	40.926
11	23.562	40.195	58.880

Following are the respective graphs for the above table,

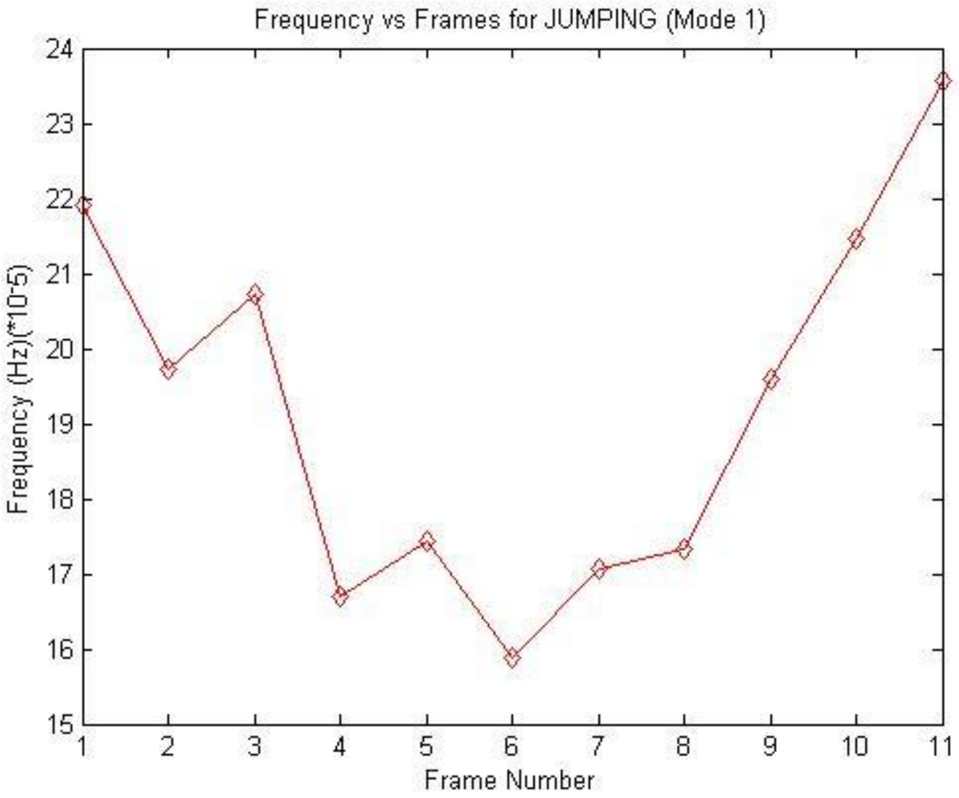


Figure 7.24 - Mode 1 Plot of Frequencies vs Frames

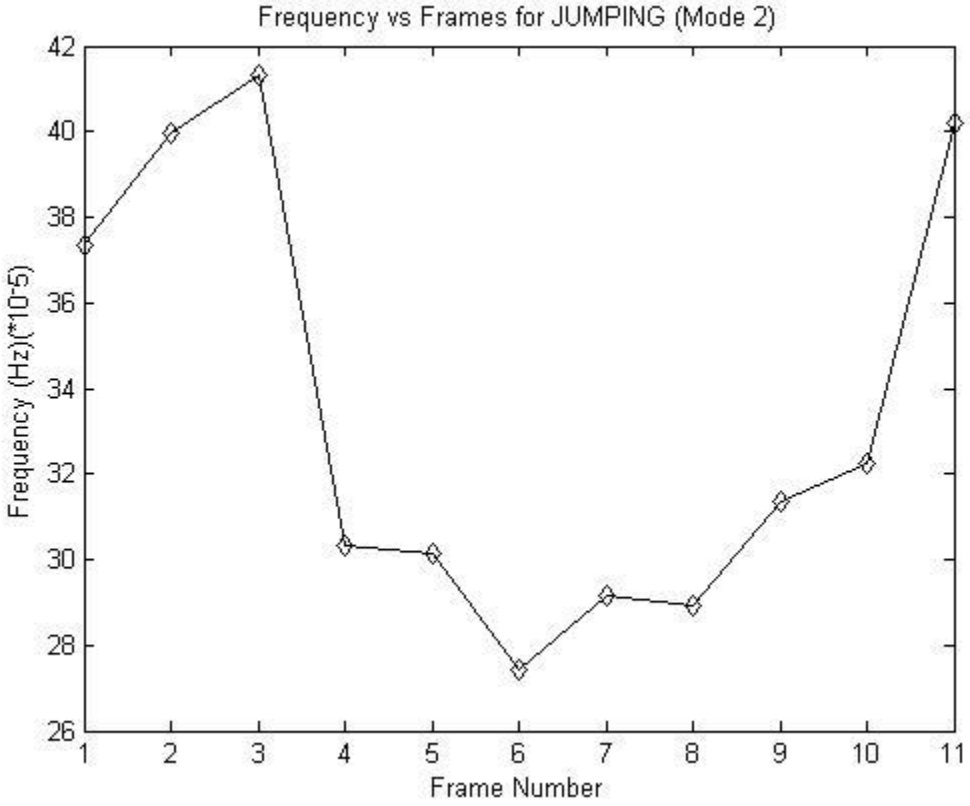


Figure 7.25 - Mode 2 Plot of Frequencies vs Frames

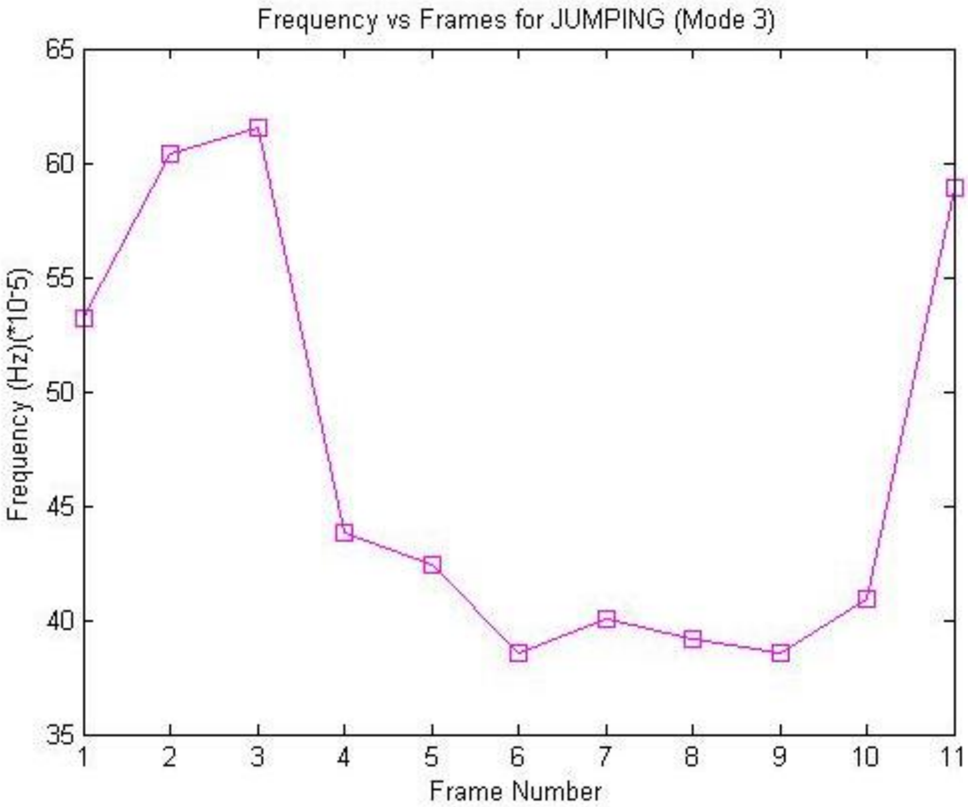


Figure 7.26 - Mode 3 Plot of Frequencies vs Frames

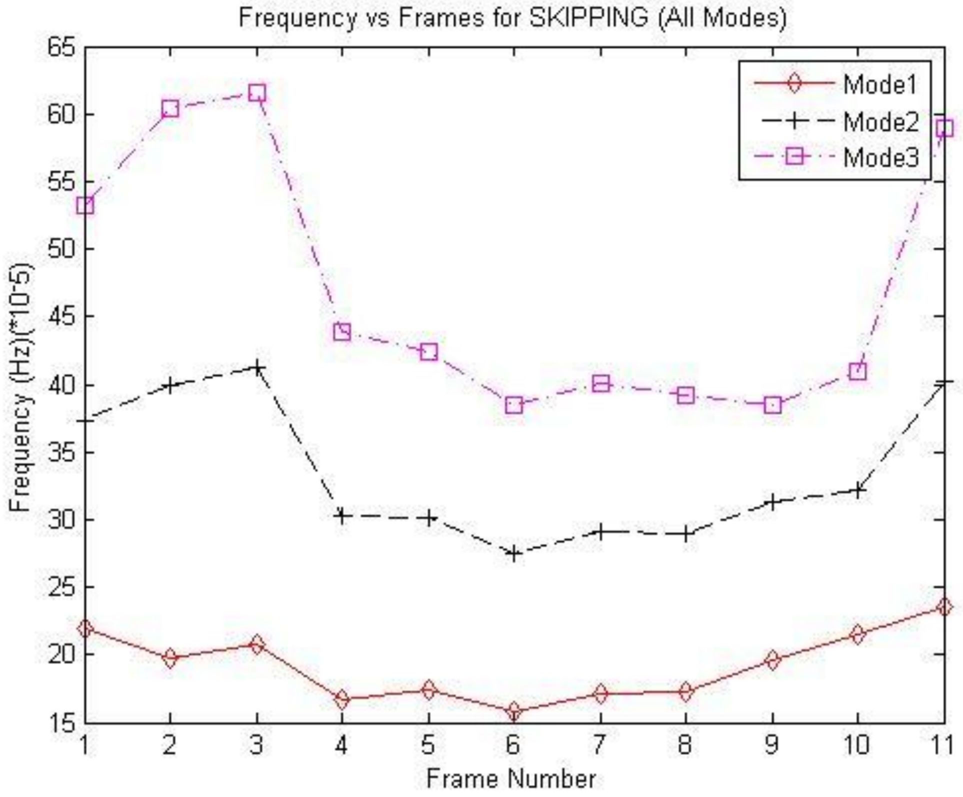


Figure 7.27 - All Modes

7.1.4HAND WAVING (ONE HAND)

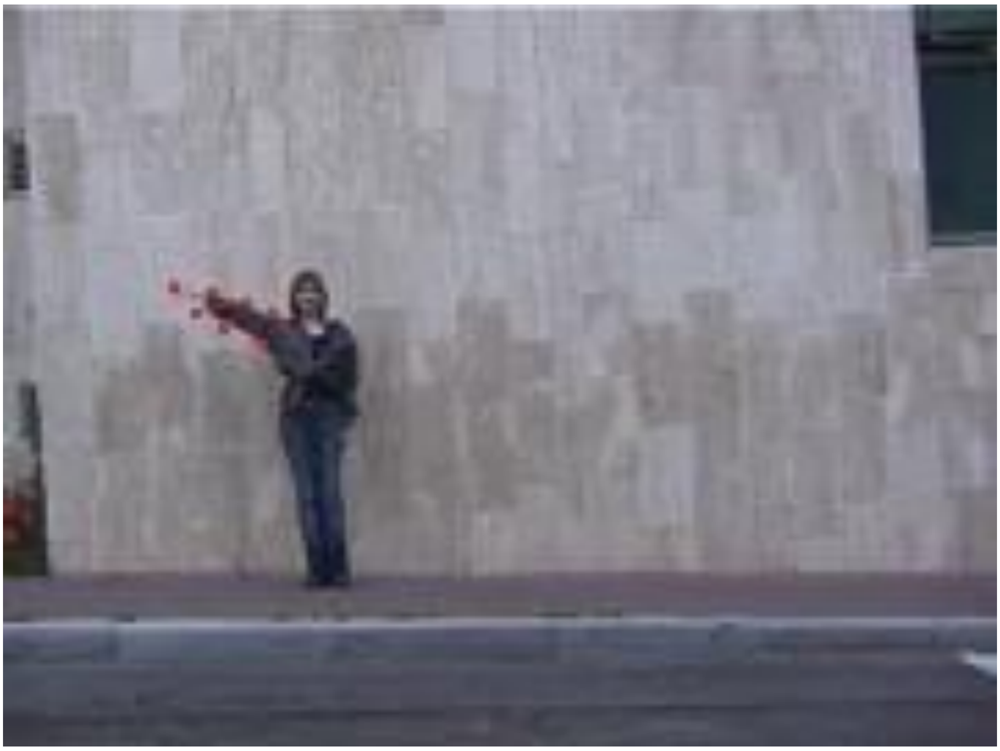


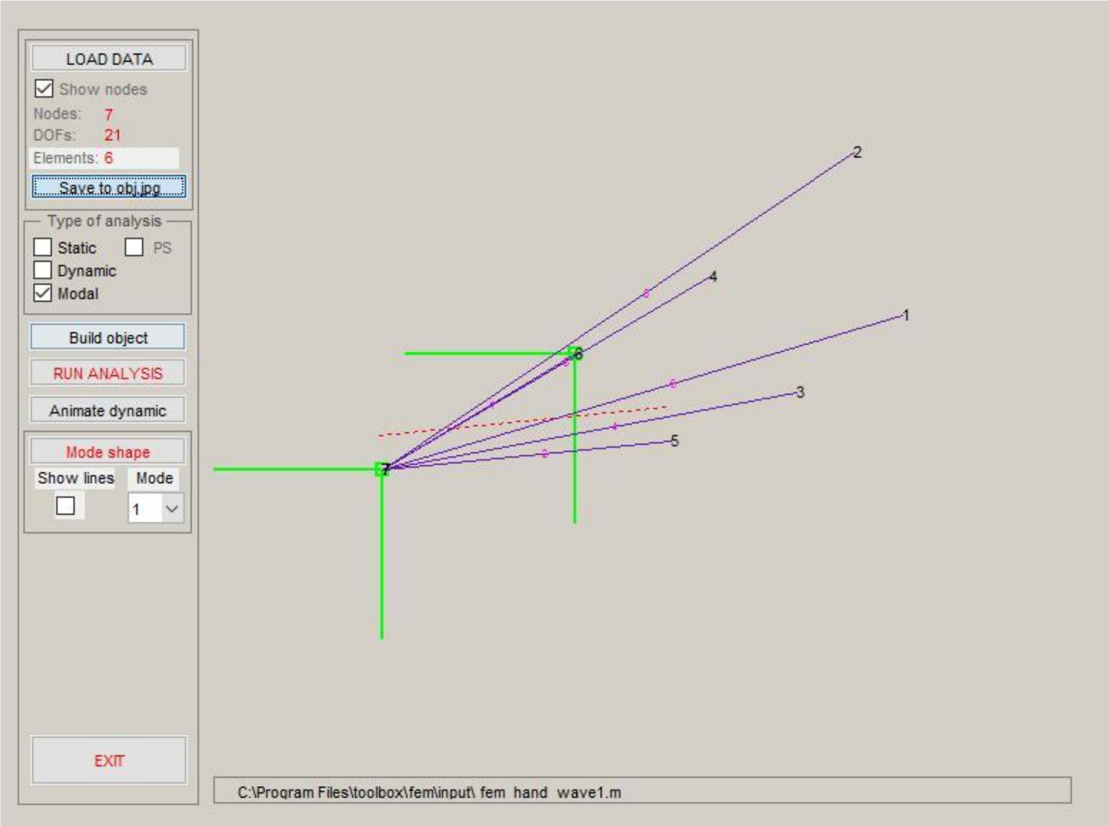
Figure 7.28 - Hand Waving (One Hand) Image Frame (Frame 5)

Nodal values of the above frame are given below

Table 7.7 Nodal values of Frame 5

S.No	X	Y
1	48.96	60.96
2	47.42	66.19
3	45.57	58.50
4	42.80	62.19
5	41.57	56.96
6	38.50	59.73
7	32.34	56.03

Forming elements and inputting the data to the toolbox, we get,



☑ Show nodes

Nodes: 7

DOFs: 21

Elements: 6

Type of analysis

□ Static □ PS

□ Dynamic

☑ Modal

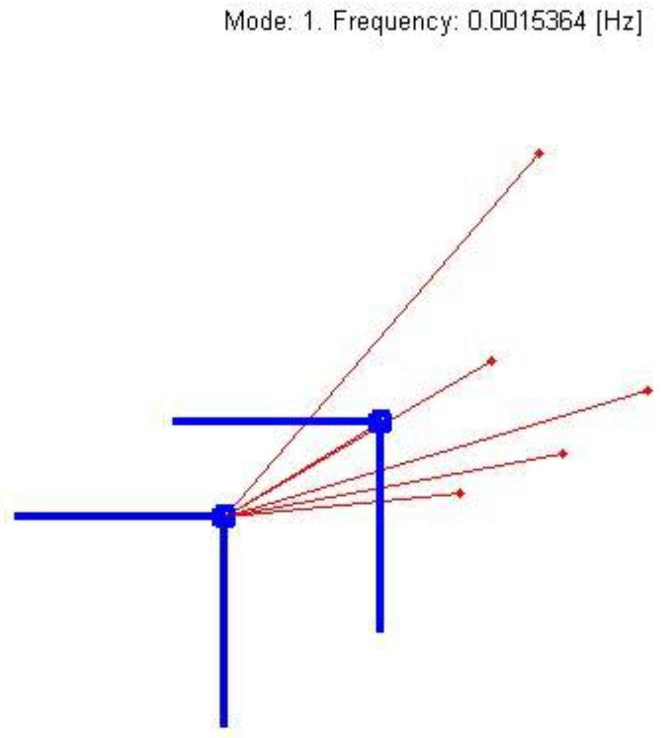
Show lines □

Mode 1

C:\Program Files\toolbox\fem\input\ fem_hand_wave1.m

Figure 7.29 - Nodes & Elements of Frame 5

Implementing the above image frame on the toolbox, we get,



Mode: 1. Frequency: 0.0015364 [Hz]

Figure 7.30 - Mode 1 Frequency for Frame 5

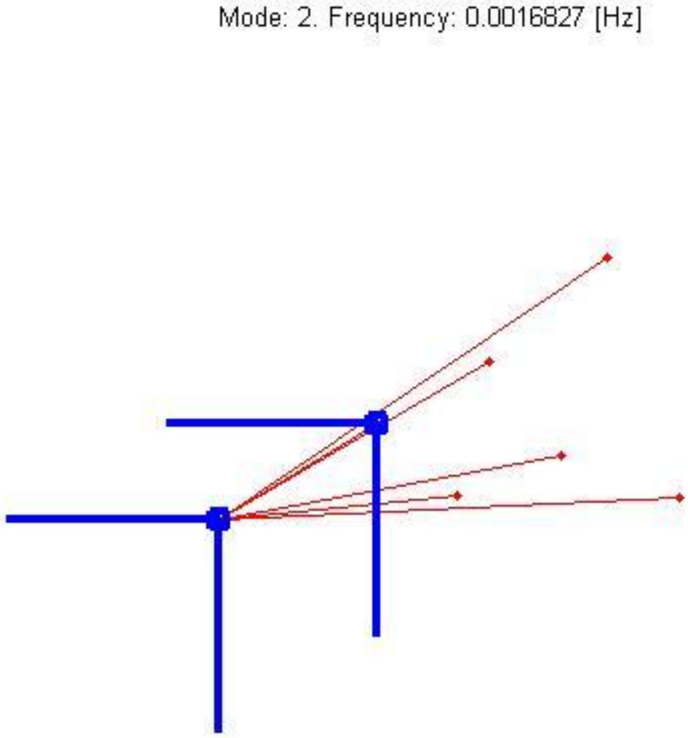


Figure 7.31 - Mode 2 Frequency for Frame 5

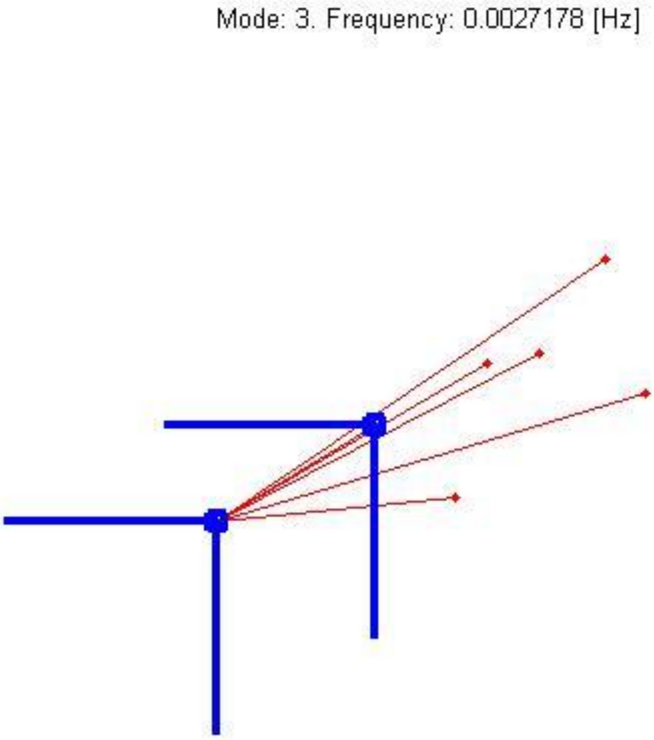


Figure 7.32 - Mode 3 Frequency for Frame 5

Calculating the modal frequencies of all such frames for one single action, we obtain the following table

Table 7.8 Modal frequencies of all frames

WAVING ONE HAND (11 FRAMES)			
FRAME NO.	MODE 1	MODE 2	MODE 3
1	120.88	170.84	227.62
2	131.96	153.81	261.73
3	144.29	165.34	322.15
4	149.53	151.82	288.93
5	153.64	168.27	271.78
6	150.85	173.90	345.38
7	131.83	138.25	263.72
8	127.42	150.94	250.82
9	116.62	133.46	249.35
10	111.09	139.45	220.71
11	110.04	135.08	186.57

Plotting these frequency values vs frames will confirm whether the action is periodic or not.

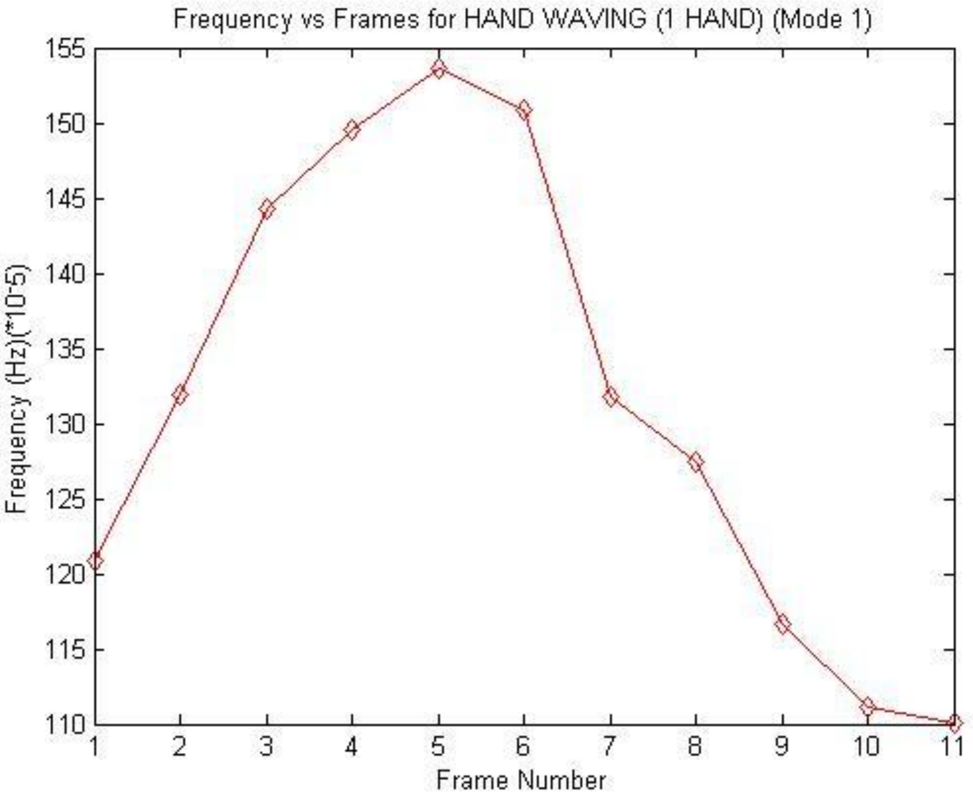


Figure 7.33 - Mode 1 Plot of Frequencies vs Frames

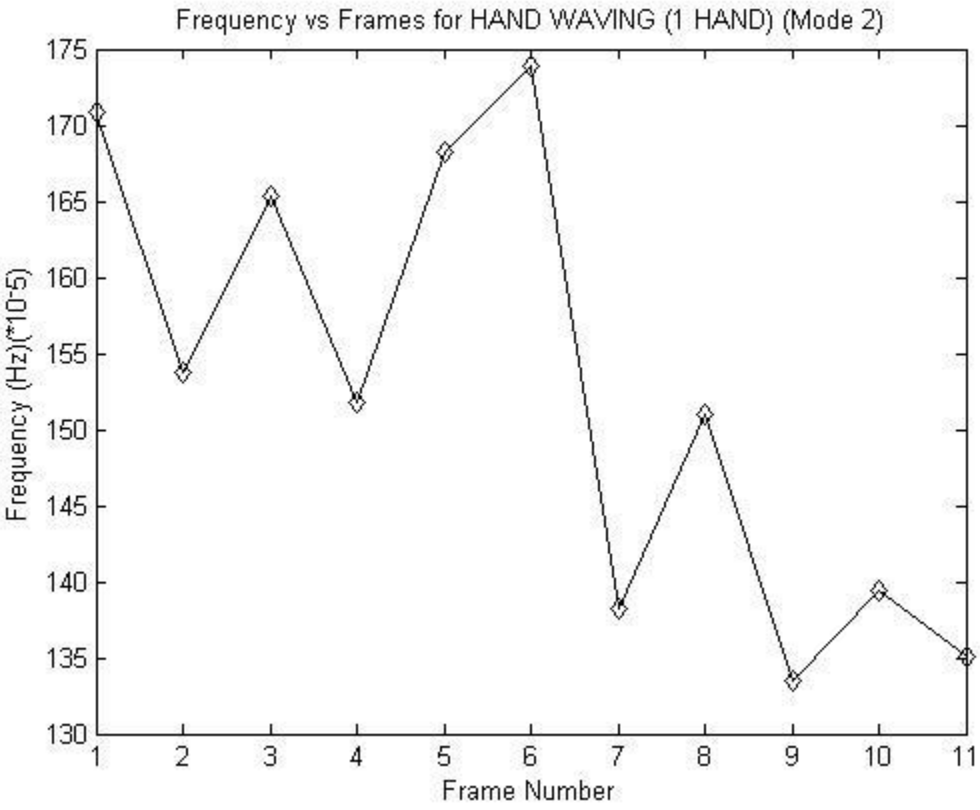


Figure 7.34 - Mode 2 Plot of Frequencies vs Frames

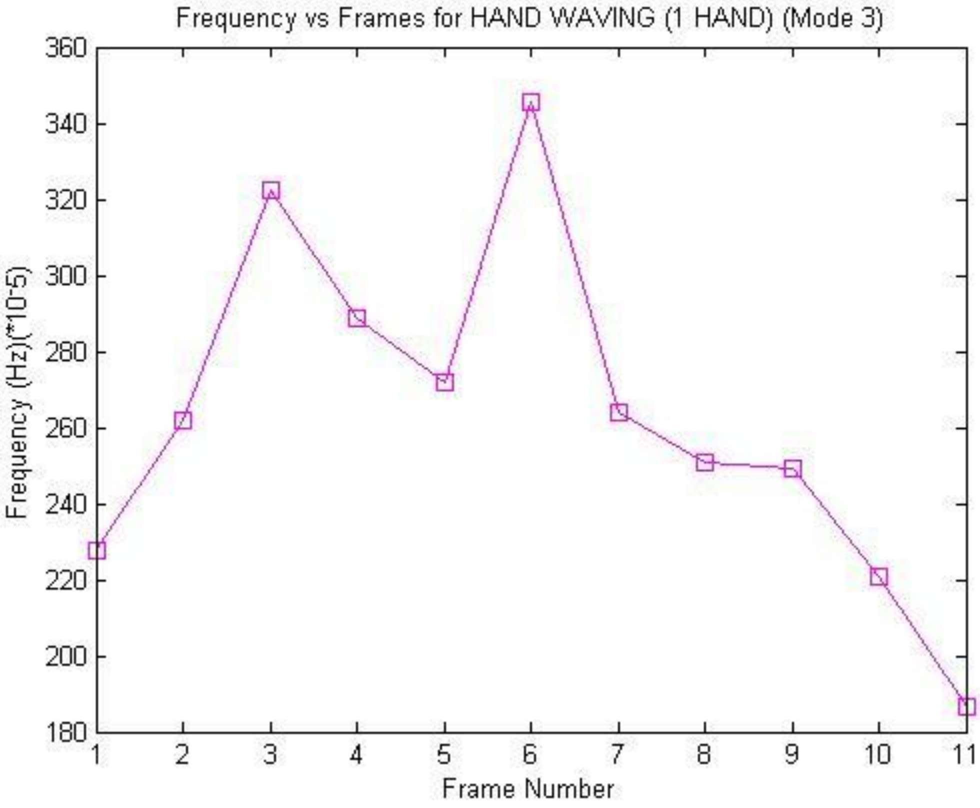


Figure 7.35 - Mode 3 Plot of Frequencies vs Frames

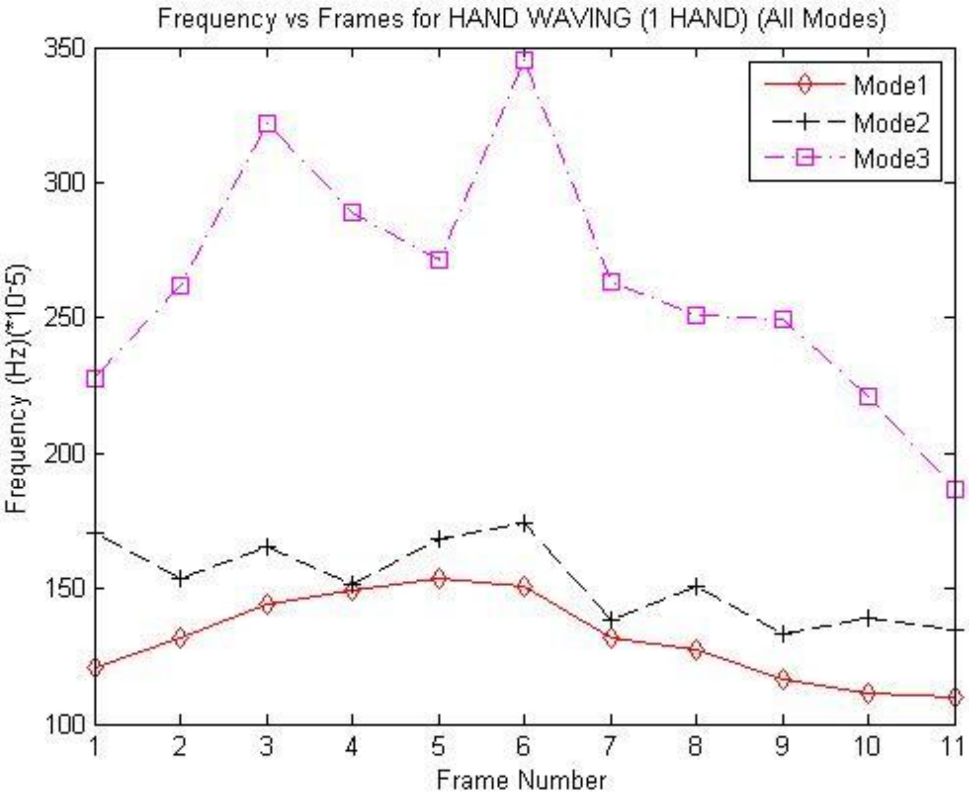


Figure 7.36 - All Modes

Hence proved, that hand waving is periodic, although mode 2 and mode 3 plots are slightly distorted.

7.1.5HAND WAVING (TWO HANDS)

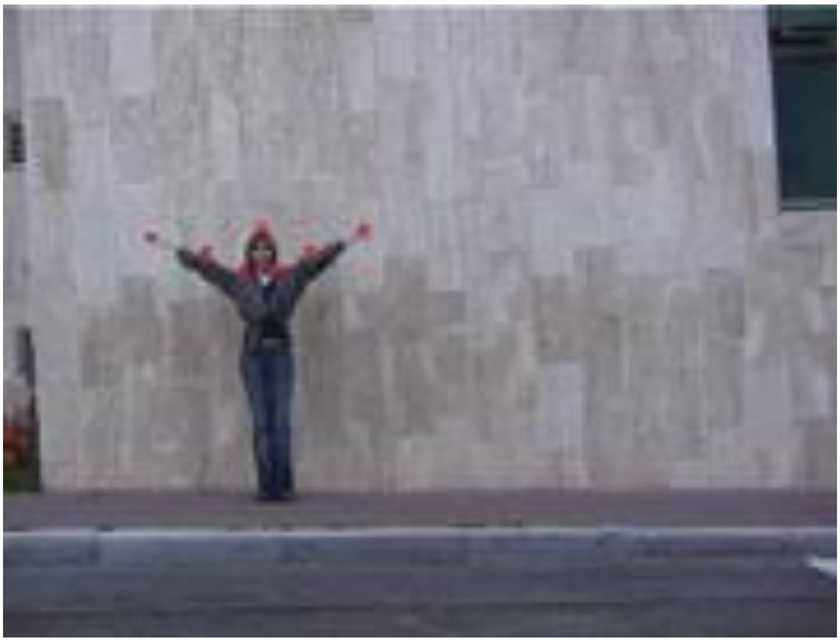


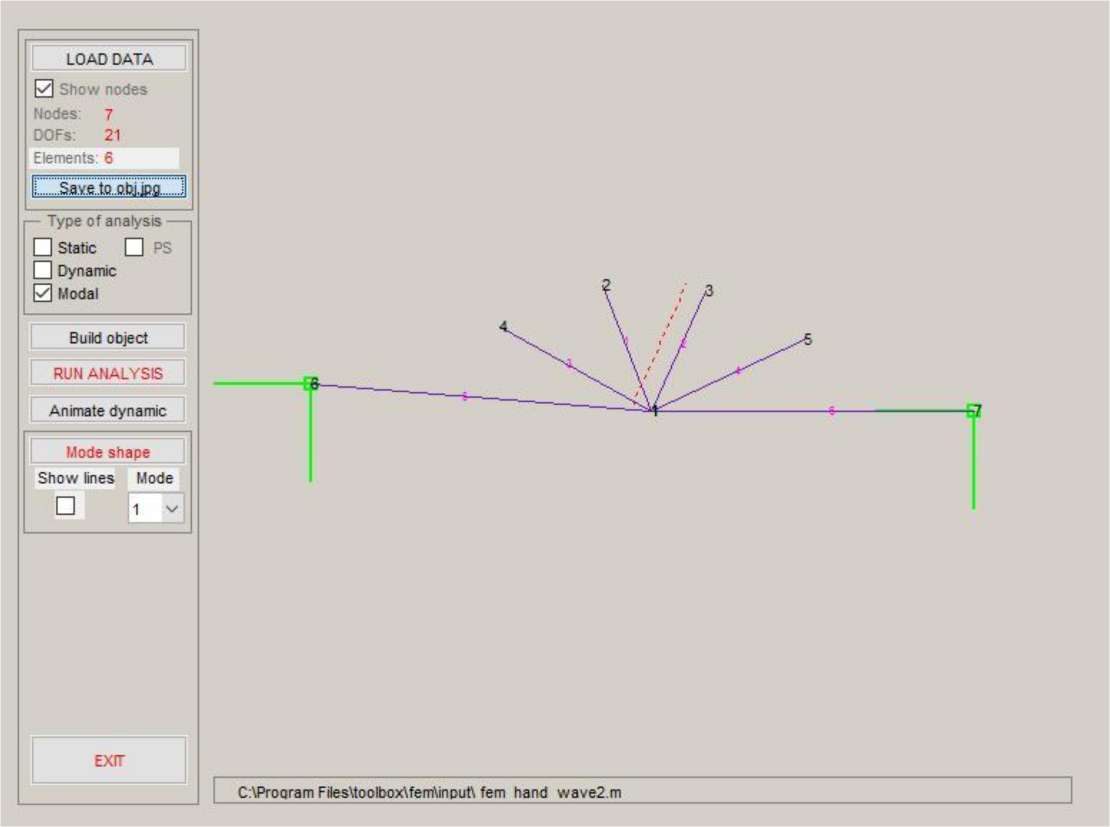
Figure 7.37 - Hand Waving (Two Hands) Image Frame (Frame 6)

In the above image, the red dots represent the nodal values, which have coordinates given in the table below

Table 7.9 Nodal values of Frame 6

S.No	X	Y
1	56.34	52.34
2	52.96	60.96
3	60.03	60.65
4	45.88	58.19
5	66.80	57.26
6	32.96	54.19
7	78.50	52.34

When we give these input values to the code of our toolbox, we get something which is shown in the following figure



LOAD DATA

☑ Show nodes

Nodes: 7

DOFs: 21

Elements: 6

Save to obj.jpg

Type of analysis

□ Static □ PS

□ Dynamic

☑ Modal

Build object

RUN ANALYSIS

Animate dynamic

Mode shape

Show lines □

Mode 1

EXIT

C:\Program Files\toolbox\fem\input\ fem_hand_wave2.m

Figure 7.38 - Nodes & Elements of Frame 6

Implementing this frame in three modes gives us the following results

Mode: 1. Frequency: 0.00091773 [Hz]

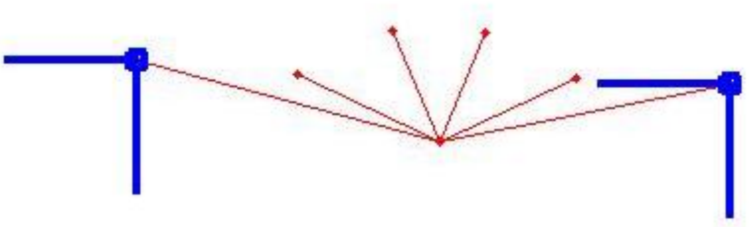


Figure 7.39 - Mode 1 Frequency for Frame 6

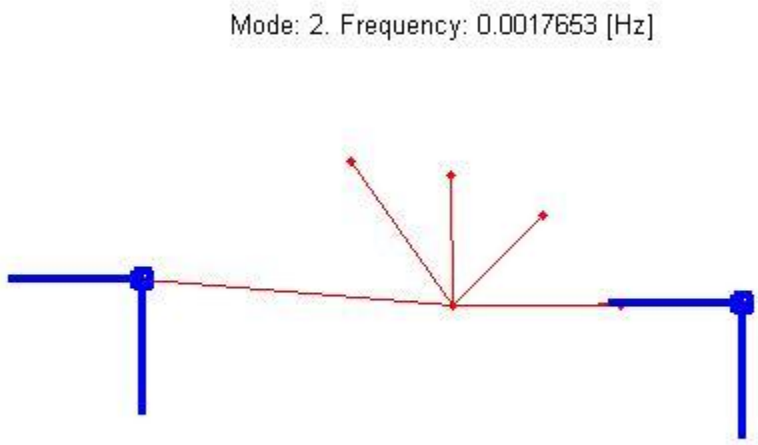


Figure 7.40 - Mode 2 Frequency for Frame 6

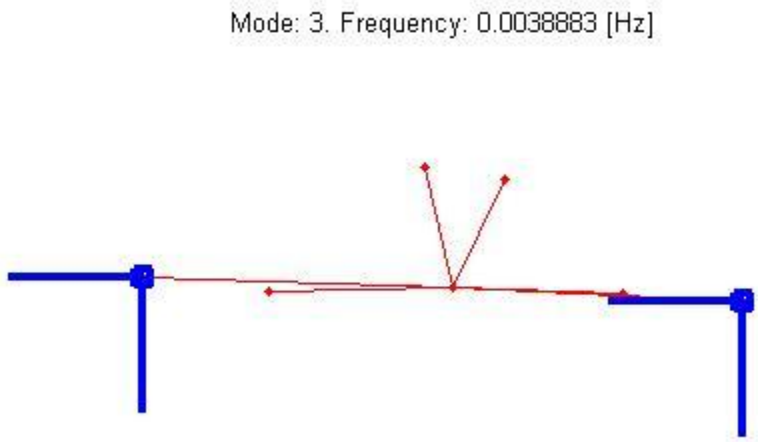


Figure 7.41 - Mode 3 Frequency for Frame 6

In a similar fashion, we work out frequencies of all the frames which are tabulated below

Table 7.10 Modal Frequencies of all Frames

WAVING TWO HANDS (13 FRAMES)			
FRAME NO.	MODE 1	MODE 2	MODE 3
1	290.980	546.150	551.170
2	305.360	535.510	562.120
3	292.460	502.610	568.700
4	248.720	306.100	525.960
5	171.530	214.340	463.230
6	91.773	176.530	388.830
7	71.130	146.110	314.010
8	72.150	146.770	341.670
9	92.839	175.740	378.710
10	212.300	238.200	446.140
11	273.300	415.440	520.590
12	304.170	516.470	542.430
13	325.920	551.240	593.670

Now, these modal frequencies are plotted as shown in the figures below

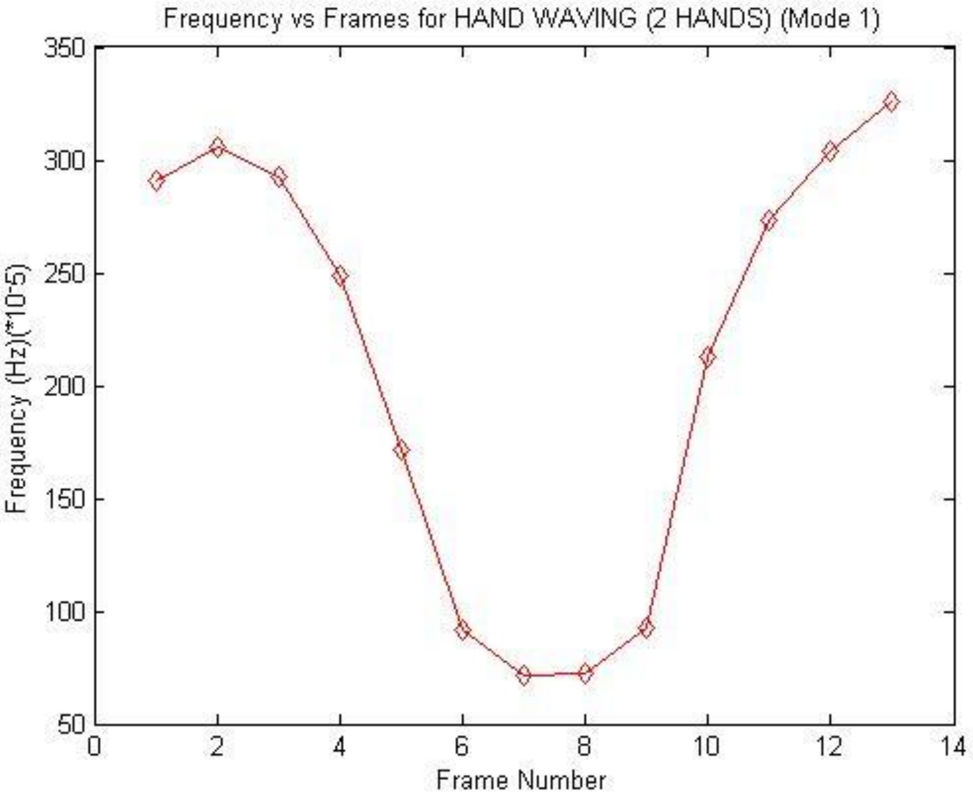


Figure 7.42 - Mode 1 Plot of Frequencies vs Frames

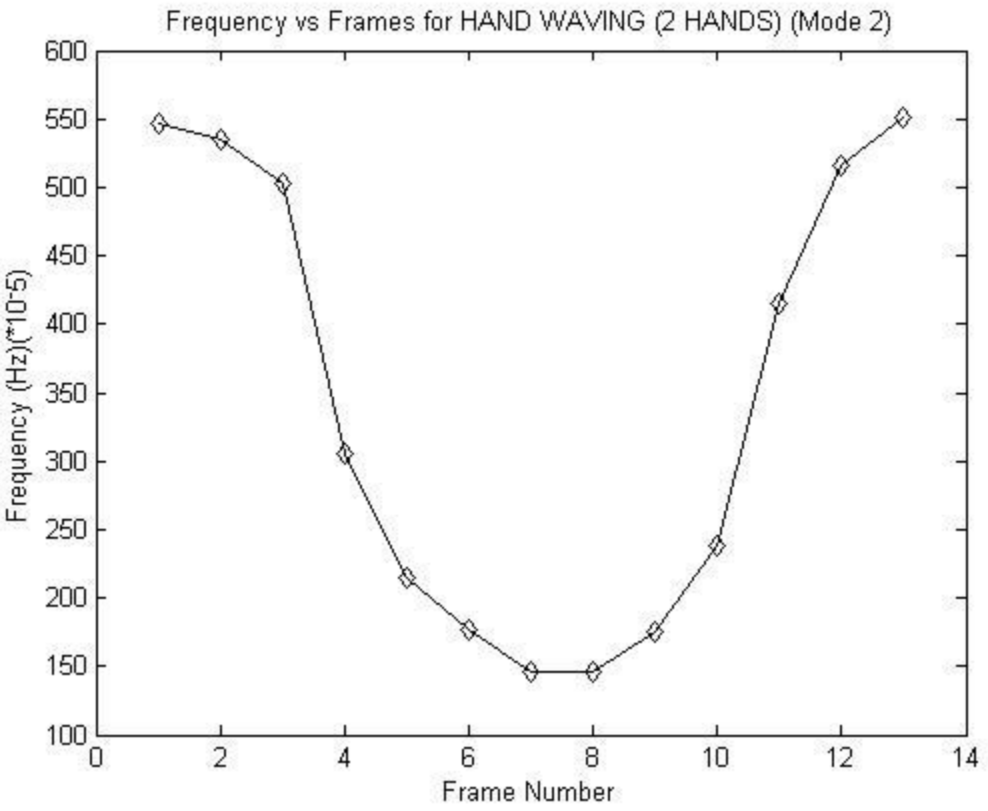


Figure 7.43 - Mode 2 Plot of Frequencies vs Frames

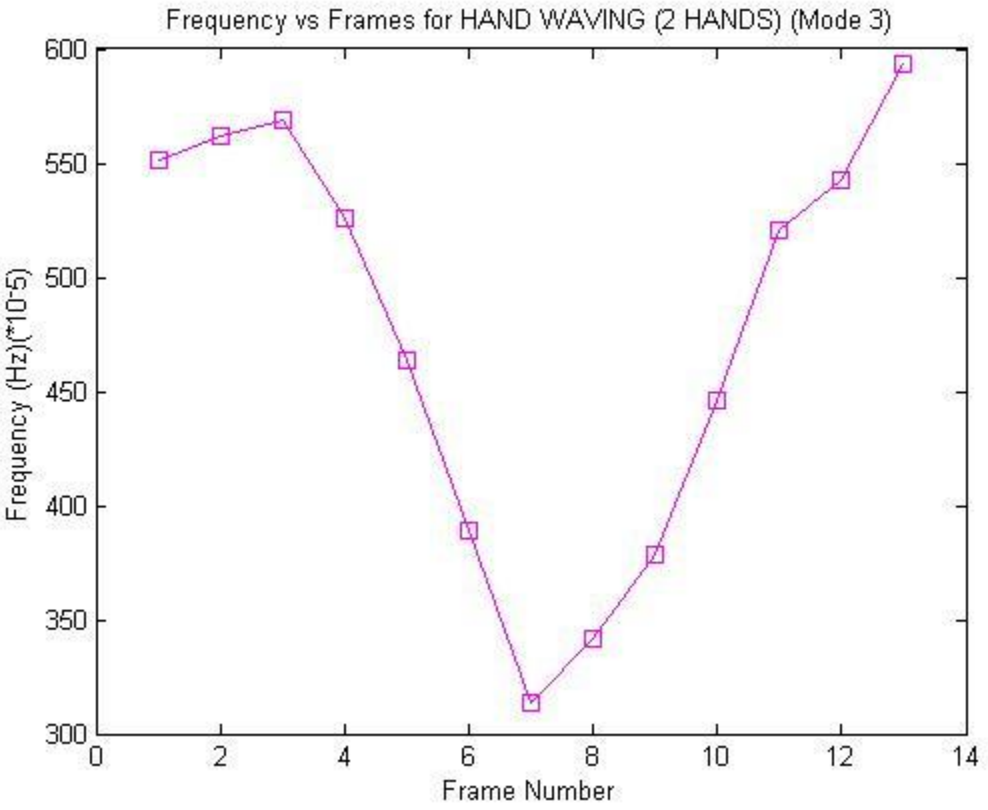


Figure 7.44 - Mode 3 Plot of Frequencies vs Frames

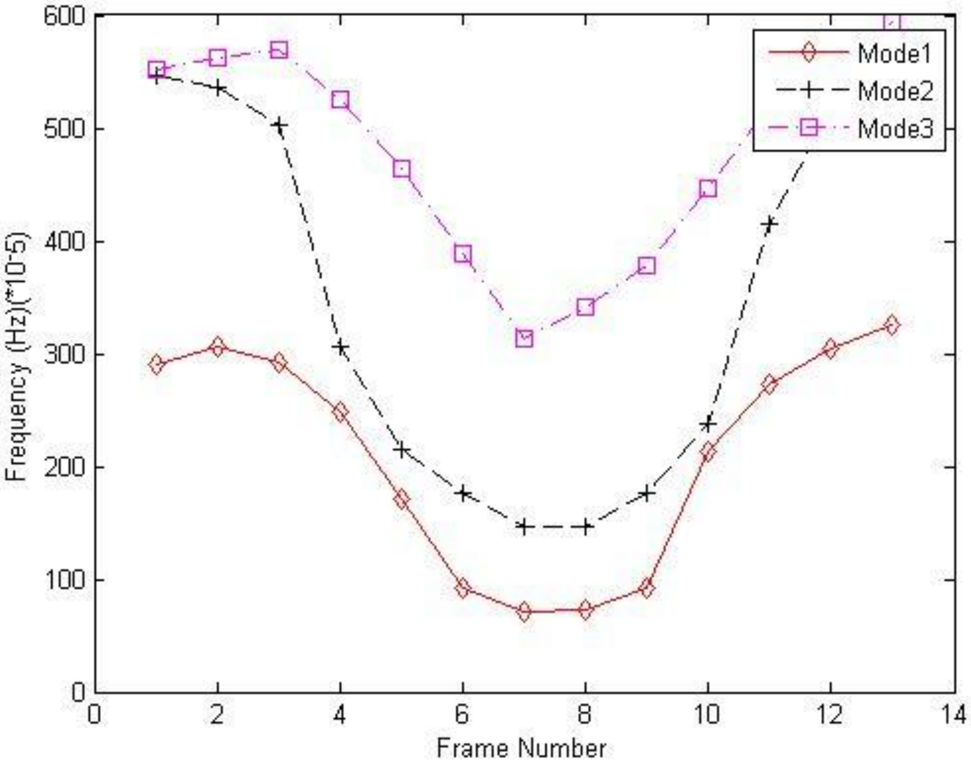


Figure 7.45 - All Modes

These graphs show that the action; hand waving with 2 hands is also periodic.

7.1.6 HOPPING (JUMPING)

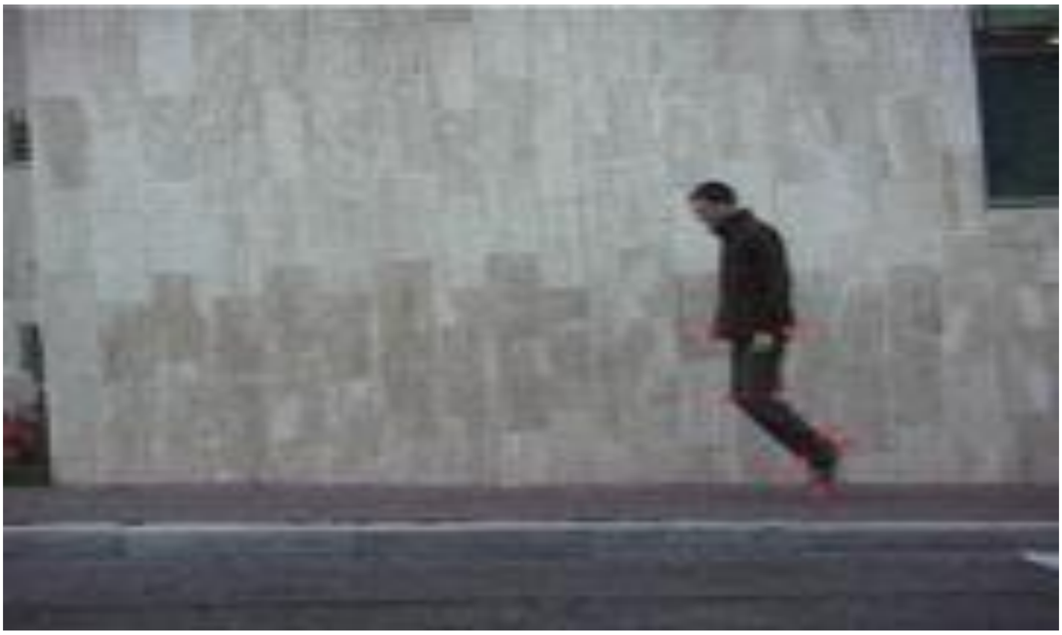
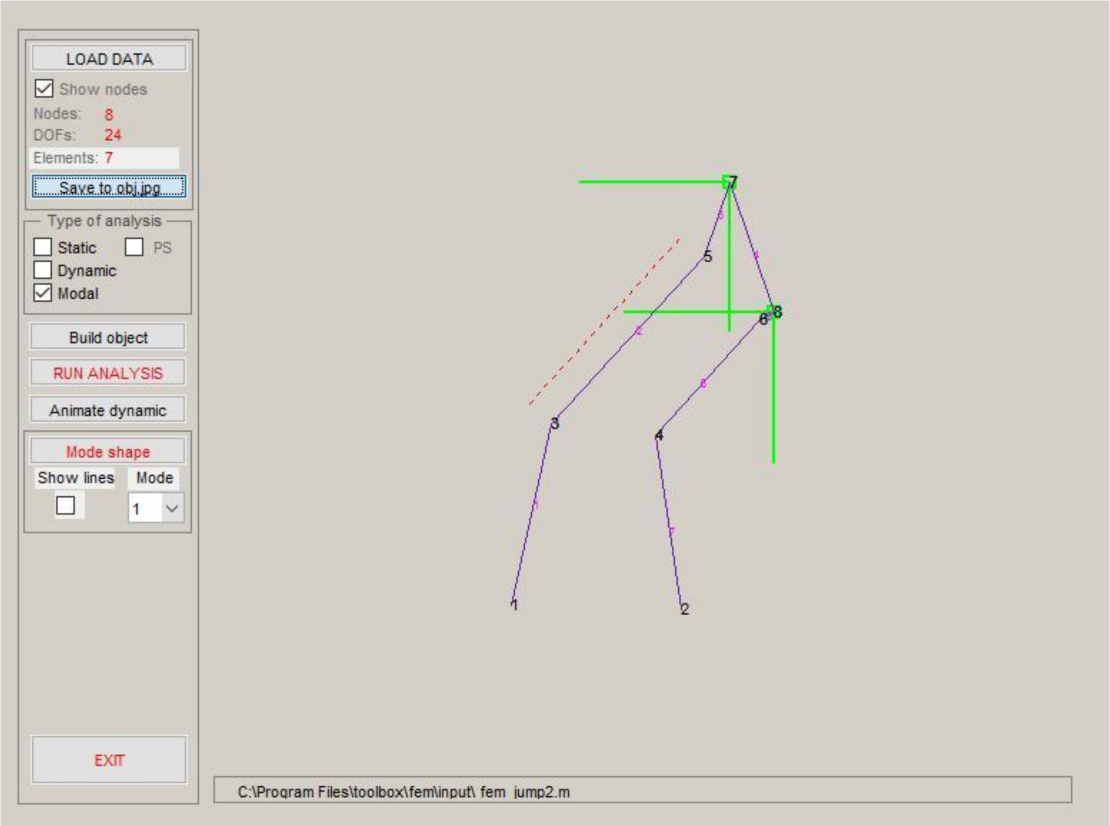


Figure 7.46 - Hopping Image Frame (Frame 8)

Table 7.11 Nodal values of Frame 8

S.No	X	Y
1	121.26	77.26
2	135.42	76.96
3	124.65	92.34
4	133.26	91.42
5	137.26	106.19
6	141.88	100.96
7	139.42	112.34
8	143.11	101.57



☑ Show nodes

Nodes: 8

DOFs: 24

Elements: 7

Type of analysis

□ Static □ PS

□ Dynamic

☑ Modal

Show lines □

Mode 1

C:\Program Files\toolbox\fem\input\ fem_jump2.m

Figure 7.47 - Nodes & Elements of Frame 8

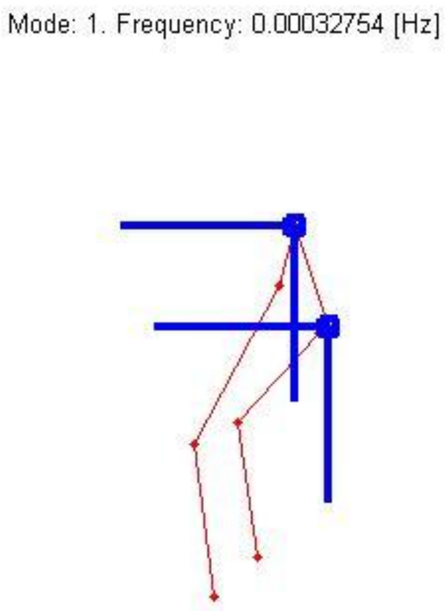

Mode: 1. Frequency: 0.00032754 [Hz]

Figure 7.48 - Mode 1 Frequency for Frame 8

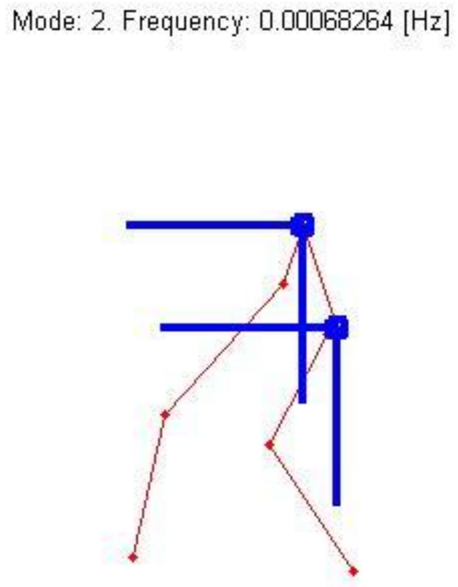


Figure 7.49 - Mode 2 Frequency for Frame 8

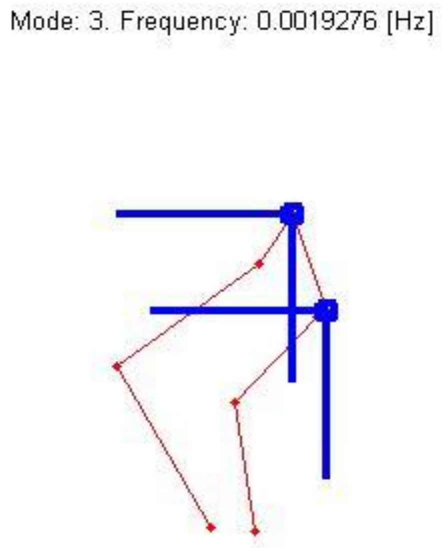


Figure 7.50 - Mode 3 Frequency for Frame 8

Table 7.12 Modal frequencies of all the Frames

JUMPING (HOPPING)			
FRAME NO.	MODE 1	MODE 2	MODE 3
1	40.809	45.787	230.430
2	38.421	47.880	187.010
3	42.559	49.809	224.030
4	37.806	46.249	226.73
5	35.427	48.499	215.31
6	31.216	44.011	193
7	30.117	61.59	184.34
8	32.754	68.264	192.76
9	38.217	73.352	226.66
10	43.797	61.592	239.58
11	45.975	64.857	276.27
12	40.215	42.211	217.96
13	39.29	42.867	244.84
14	43.412	47.506	262.8
15	44.569	50.075	218.09
16	49.481	53.242	212.28
17	46.061	51.967	208.7

Plotting these frequencies, we get the following results

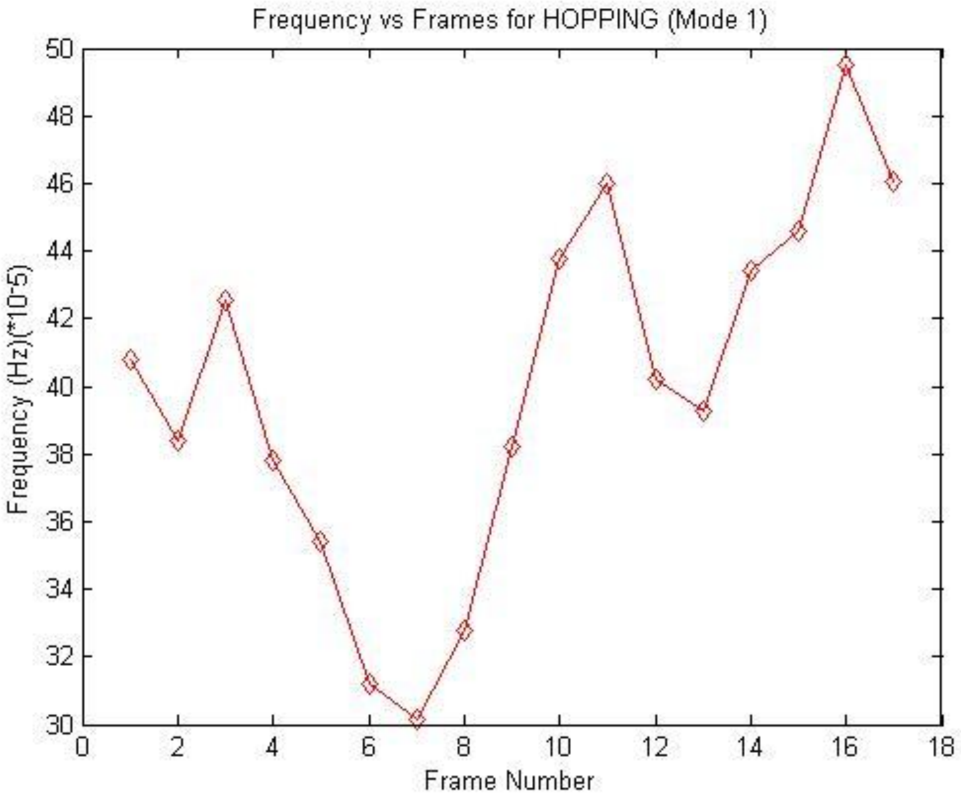


Figure 7.51 - Mode 1 Plot of Frequencies vs Frames

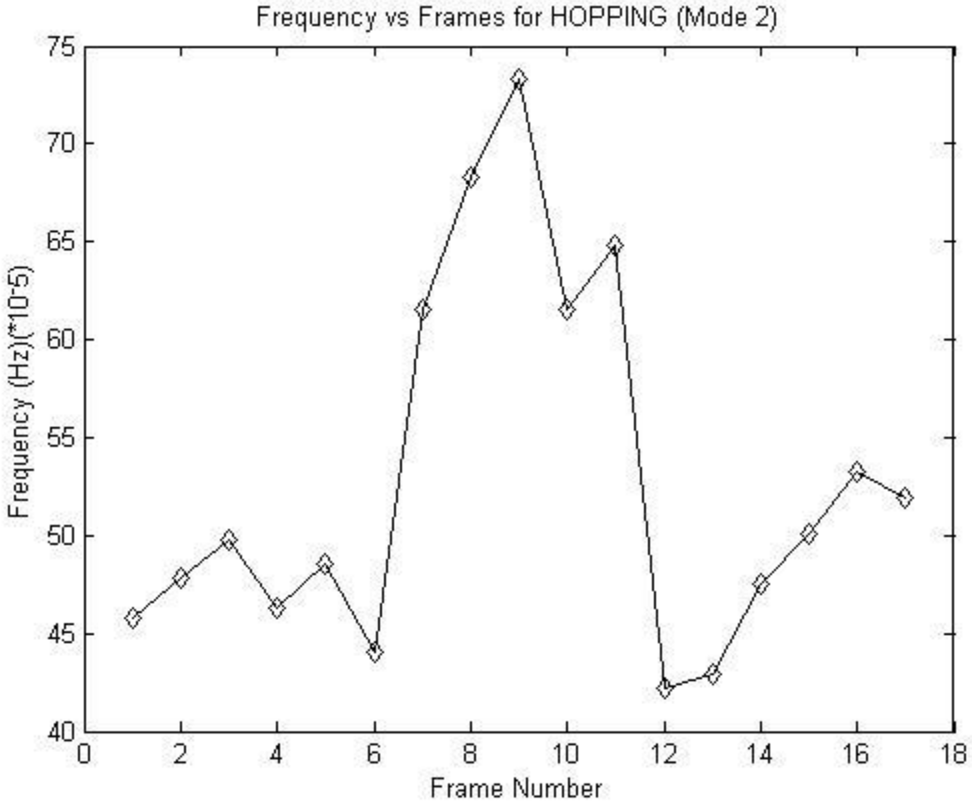


Figure 7.52 - Mode 2 Plot of Frequencies vs Frames

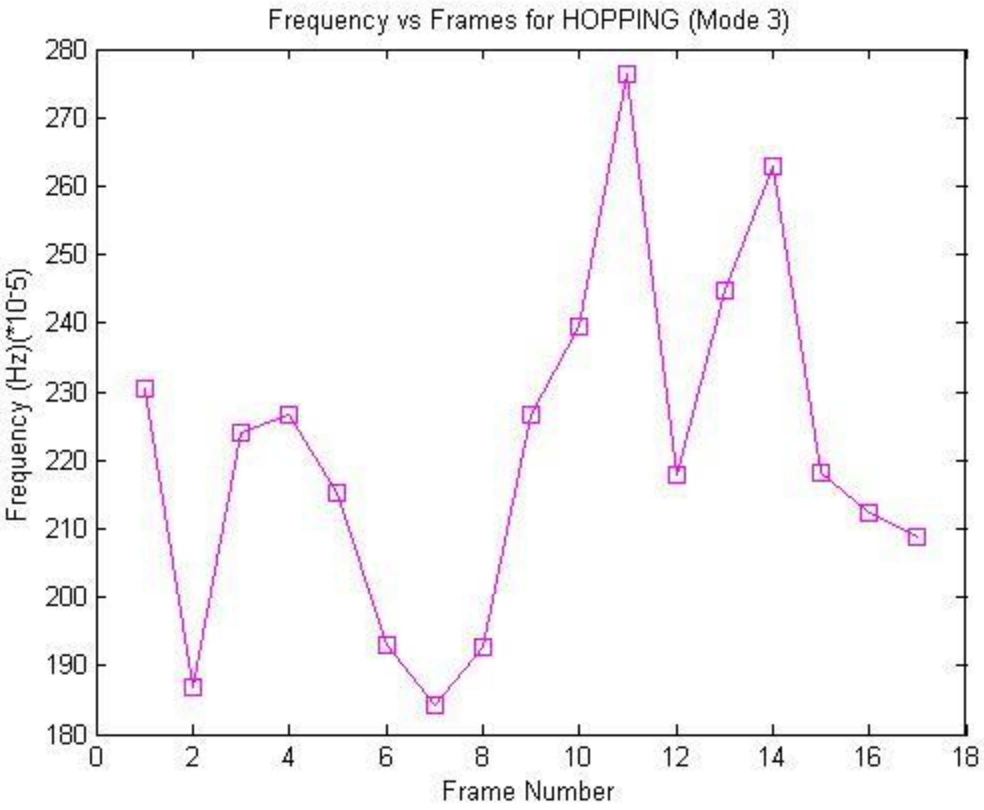


Figure 7.53 - Mode 3 Plot of Frequencies vs Frames

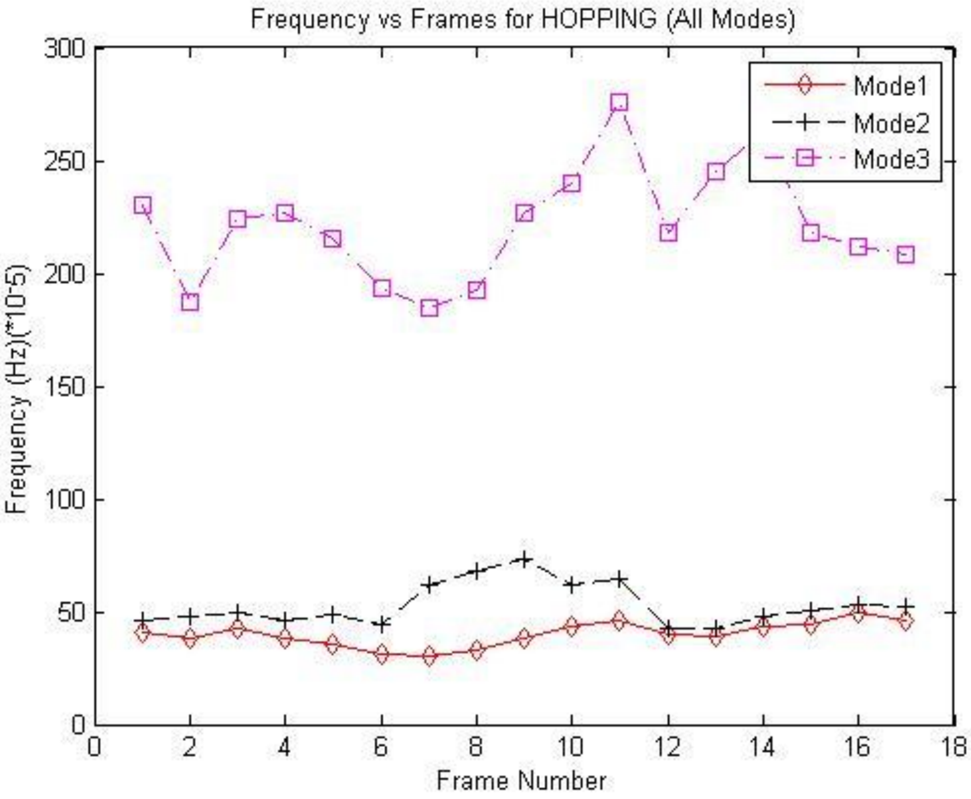


Figure 7.54 - All Modes

INFERENCE

From the above analysis and calculations, we can see that some actions are periodic and some actions are not periodic. Also, among the periodic actions, different actions have different periodicity. So, in the above implementation we can distinguish between actions on the basis of the following two factors :

1) On the basis of periodic and non-periodic actions. Clapping, Jumping, Hand waving (with one or two hands) are periodic actions, whereas skipping and hopping are non-periodic actions(on the basis of the results obtained above).

2) On the basis of different periodicity. Different periodic actions have different periodicity.

7.2 ANOMALY DETECTION

In this section, we make use of the FEM toolbox in order to detect an unusual activity in a video. This is done by considering 2 frames of a video and implementing them in the toolbox to get frequency values. One frame will be normal and the second frame will contain information of an unusual activity, which will result in 2 different frequencies for the two respective frames. Let us consider the following example.

The image given below is of the normal frame where there is nothing but water surface.

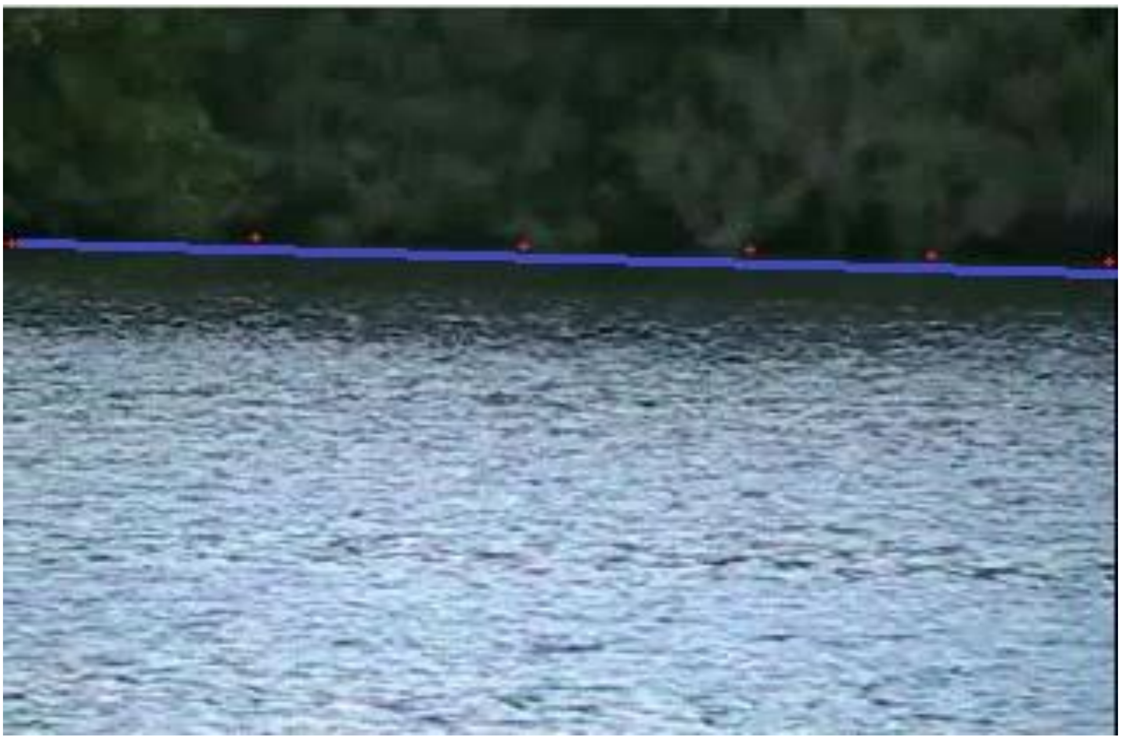


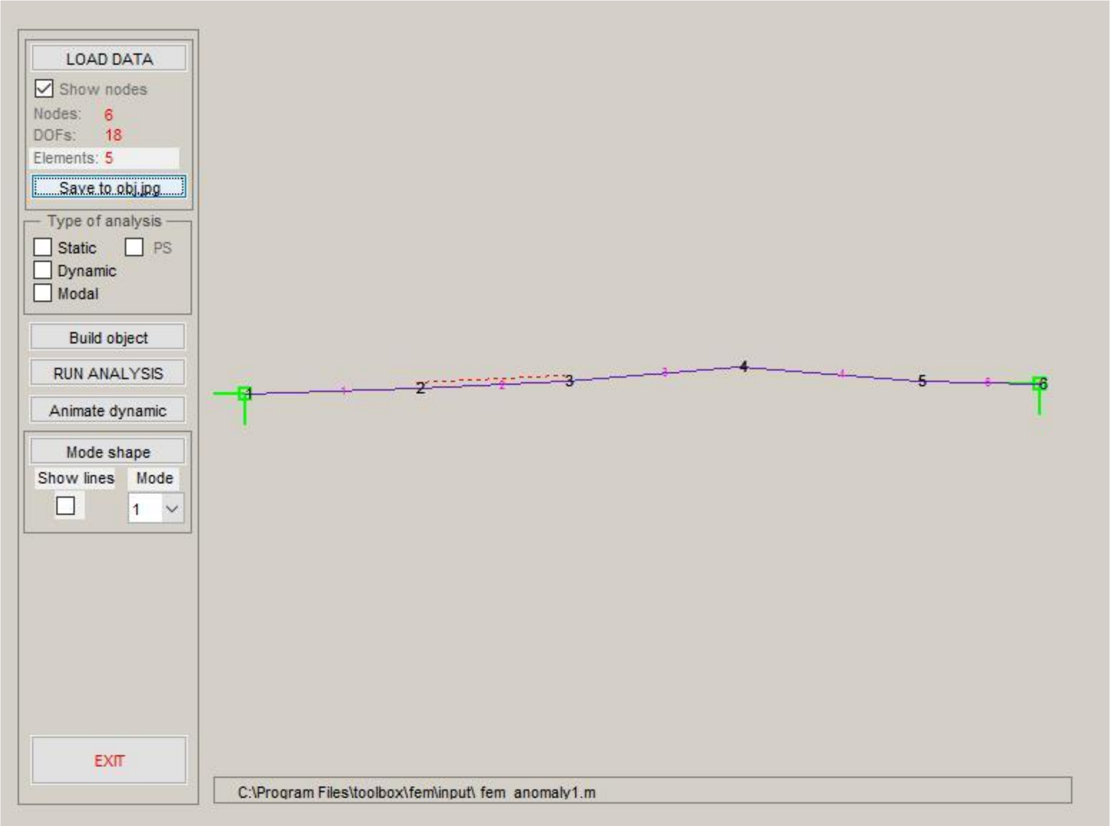
Figure 7.55 - Normal Image Frame

The red dots represent the nodes and the blue line is the division between the water surface and the ground. The coordinates of the nodes are tabulated in the table below.

Table 7.13 Nodal values of the normal frame

S.No	X	Y
1	121.26	77.26
2	135.42	76.96

3	124.65	92.34
4	133.26	91.42
5	137.26	106.19
6	141.88	100.96



☑ Show nodes
Nodes: 6
DOFs: 18
Elements: 5

Type of analysis
□ Static □ PS
□ Dynamic
□ Modal

Show lines □
Mode 1

C:\Program Files\toolbox\fem\input\ fem anomaly1.m

Figure 7.56 - Nodes & Elements of the normal frame

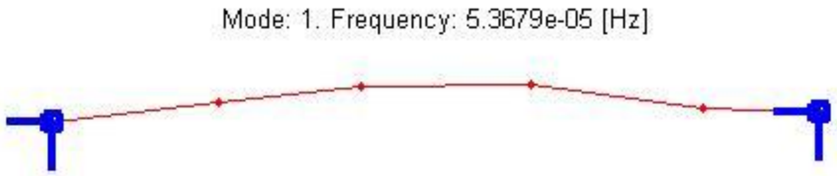


*Figure 7.57 - Modal Frequency of the normal frame : 5.3679 * 10$^{(-5)}$*

Now, we have the abnormal frame where there is a boat. The red dots are the nodes and the brown line joins those nodes to form elements.

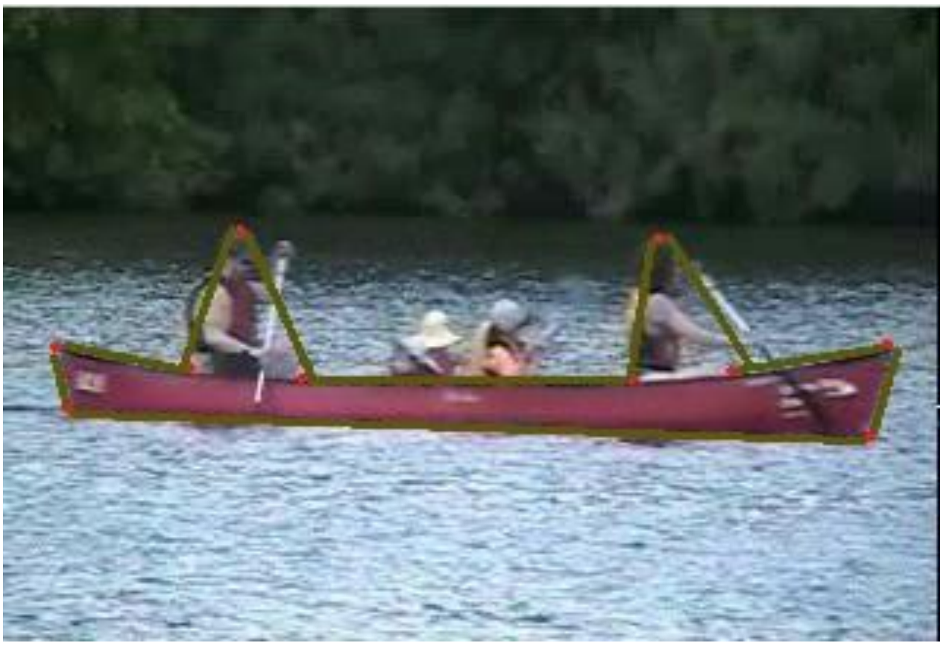
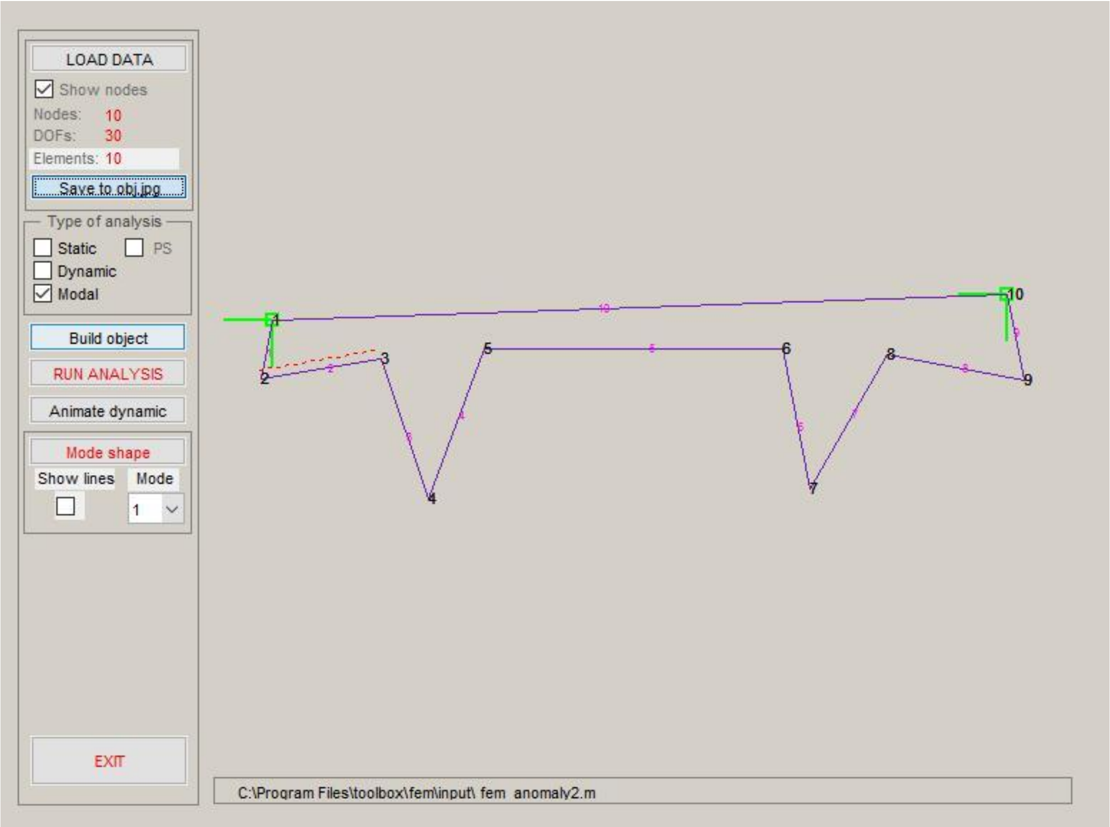


Figure 7.58 - Frame with Abnormality

Table 7.14 Nodal values of the abnormal frame

S.No	X	Y
1	23.14	151.63
2	18.99	129.36
3	63.51	136.91
4	81.25	84.83
5	102.00	140.68
6	212.95	140.68
7	223.14	88.61
8	251.82	138.80
9	302.76	128.99
10	296.34	161.06



☑ Show nodes

Nodes: 10

DOFs: 30

Elements: 10

□ Static □ PS

□ Dynamic

☑ Modal

□ Mode 1

C:\Program Files\toolbox\fem\input\ fem anomaly2.m

Figure 7.59 - Implementing the Abnormal frame in FEM

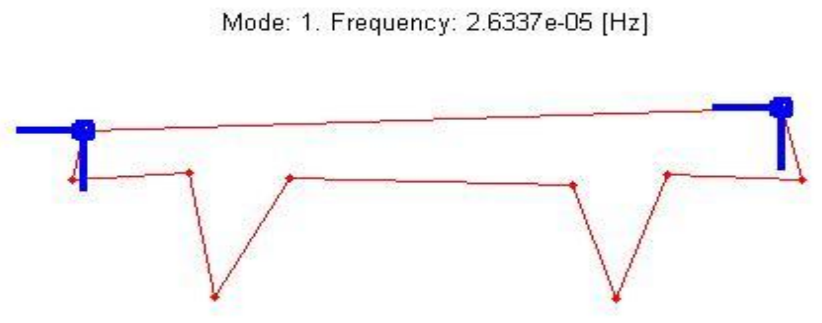


*Figure 7.60 - Modal Frequency of the abnormal frame : 2.6337 * $10^{(-5)}$*

As we can see, the two frames, when feeded into the FEM toolbox have given entirely different frequencies.

Normal Frame : 5.3679 * $10^{(-5)}$Hz

Abnormal Frame : 2.6337 * $10^{(-5)}$Hz

Therefore, we can easily show that changes in these frequencies indicates a change in the video scenario, to be more specific, an anomaly.

CHAPTER 8 CONCLUSION & FUTURE WORK

In this thesis, we implemented video sequences (to be specific, video frames) using the FEM toolbox. This method successfully differentiated between two distinct actions as well as a normal and an abnormal scene; by giving out modal frequencies as an output. This is a very novel approach as a toolbox meant for the field of solid mechanics has been used for a task like image processing. But in this method, videos are implemented on a small scale (using less number of frames per action or per video), so implementing it on a large scale, i.e., heavy videos, is something that can be considered as future work in this field.

CHAPTER 9 REFERENCES

1) iwringer.wordpress.com
2) Patcha, A.. "An overview of anomaly detectiontechniques: Existing solutions and latesttechnological trends", Computer Networks,20070822.
3) Li, Ce, Zhenjun Han, Qixiang Ye, and JianbinJiao. "Visual abnormal behavior detectionbased on trajectory sparse reconstructionanalysis", Neurocomputing, 2013.
4) Tao Xiang. "IEEE transactions on Pattern Analysis and Machine Intelligence, 5/2008.
5) Xudong Zhu. "Human behavior clustering foranomaly detection", Frontiers of ComputerScience in China, 04/28/2011
6) Xiong, G.. "An energy model approach topeople counting for abnormal crowd behaviordetection", Neurocomputing, 20120415.
7) codedevelopment.net
8) nptel.iitm.ac.in
9) www.datascience.com
10) ethesis.nitrkl.ac.in
11) For dataset, KTH, http://www.nada.kth.se
12) For dataset, Weizmann, http://www.wisdom.weizmann.ac.il.