UNIVERSITY OF CALIFORNIA,
IRVINE

Novel Side-Channel Attack Model for Cyber-Physical Additive Manufacturing Systems

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Electrical and Computer Engineering

by

Sujit Rokka Chhetri

Thesis Committee:
Assistant Professor Mohammad Al Faruque, Chair
Professor Pai H. Chou
Assistant Professor Aparna Chandramowlishwaran

2016

DEDICATION

To my parents and my brother

for their remarkable

belief, support, and love.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Page

ACKNOWLEDGMENTS

I would like to thank my committee chair, Professor Mohammad Abdullah Al Faruque, for constantly guiding me through the research and providing me his valuable insights. Without his excellent supervision and persistent mentoring, I would not have been able to complete this thesis.

I would also like to thank my committee members Professor Pai H. Chou and Professor Aparna Chandramowlishwaran, for providing me guidance, and giving me critical feedback when I needed the most.

I would like to thank all my colleagues with whom I have had intellectually stimulating discussions about the research matter, and my families for always providing me their unconditional love and support.

I thank IEEE for the permission to include content of my thesis, which was originally published in ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS).

I would also like to thank NSF for partially funding the project under CPS grant CNS-1546993.

ABSTRACT OF THE THESIS

Novel Side-Channel Attack Model for Cyber-Physical Additive Manufacturing Systems

By

Sujit Rokka Chhetri

Master of Science in Electrical and Computer Engineering

University of California, Irvine, 2016

Assistant Professor Mohammad Abdullah Al Faruque, Chair

Cyber-physical systems consists of tight integration of cyber and physical domain components. Due to this, they are prone to various cross-domain attacks. One form of such attacks can take place in the form of physical-to-cyber domain attacks, which can cause confidentiality breach of the system. This is due to the fact that some of the cyber-domain information manifest in terms of physical actions such as motion, temperature change, etc. These physical actions may unintentionally leak information about the cyber-domain through the side-channels. Up until now there has been no study highlighting how these form of cross-domain attack can affect the cyber-physical additive manufacturing systems. Hence, in this thesis we present the analysis of acoustic side-channel to demonstrate how it can be leverage to build novel attack model and breach the confidentiality of the additive manufacturing system (such as 3D printers). Side-channels such as acoustic, thermal, and power allow attackers to acquire the information without actually leveraging the vulnerability of the algorithms implemented in the system. In 3D printers, geometry,

process, and machine information are the intellectual properties, which are stored in the cyber domain (G-code). We have designed an attack model that consists of digital signal processing, machine learning algorithms, and context-based post processing to steal the intellectual property by reconstructing the G-code and thus the test objects. We have successfully reconstructed various test objects with an average axis prediction accuracy of 78.35% and an average length prediction error of 17.82%.

CHAPTER 1: INTRODUCTION

Cyber-Physical Systems (CPS) consist of the integration of computation, physical, and networking components [1]. The synergy of these components results in a new form of vulnerabilities, which cannot be addressed by traditional security solutions designed for the individual components. Additive manufacturing is an example of CPS, where 3D objects are created layer by layer [2]. Fused Deposition Modeling (FDM) is one of the technologies used in additive manufacturing, where plastic or metal filaments, heated slightly above their melting point, are deposited to construct a 3D object [3]. Several sectors, such as medical, automotive, and aerospace, are increasingly adopting the use of these additive manufacturing systems [4] [5] [6] [7]. In addition, agencies like the U.S. Air Force [8] , Navy [9], and NASA [10] are also incorporating additive manufacturing into their manufacturing processes. The revenue of the additive manufacturing industry is expected to exceed $21B by 2020 [11]. In fact cyber-physical additive manufacturing has been termed as one of the proponents of the next industrial revolution [12]. It is estimated that with cyber-physical system as the foundation, the industrial internet of things will create a global GDP of $15 trillion by 2030 [13], and additive manufacturing will be a major part of this fourth industrial revolution.

The promising forecast of the additive manufacturing, its application, and the revenue generated, however, hides a major challenge for its adoption in the next industrial revolution. One of the major challenge will be in securing the cyber-physical additive manufacturing system [14]. Analyzing the economic forecast for additive manufacturing, a security breach can have large financial impact on the manufacturing industry [15] [16]. In

this thesis, out of the three security requirements of a cyber-physical system, *integrity*, *confidentiality*, and *availability* [17], we will focus in the issues regarding the *confidentiality* breach. Attackers who target additive manufacturing systems will often be motivated by industrial espionage of Intellectual Property (IP) [18] [19]. The world economy relies heavily on IP-based industries, which produce and protect their designs through IP rights. In the U.S. alone, the IP-intensive industries have been known to account for 34.8% of the gross domestic product [20], and they are bound to face security issues. In fact, it has been estimated that, by 2018, 3D printing of pirated designs will result in annual IP losses of $100 billion [21].

IP in additive manufacturing consists of the internal and external structure of the object, the process parameters, and the machine specific tuning parameters [22]. To produce a 3D object, design information (which contains IP) is supplied to the manufacturing system in the form of G-code. G-code, a programming language, is primarily used in FDM to control the system components and parameters such as speed, temperature, and extrusion amount [23]. If these designs are stolen, they can be manipulated to harm the image of the company, or even worse, can cause the company to lose its IP (as it is stolen before production) [24]. Currently, IP theft mainly occurs through the cyber domain (e.g., Operation Aurora [25], GhostNet [26]) but IP information can also be leaked through the physical domain (side-channels). A common example of this is to use side-channel information (e.g., timing data, acoustics, power dissipation, and electromagnetic emission) from devices performing cryptographic computation to determine their secret keys [27].

In this thesis we highlight the possibility of physical-to-cyber domain attacks on cyber-physical additive manufacturing system, and motivate a general research interest in novel ways to minimize the side-channel leakage during design and run time. It is probably not possible to make a system completely secure [17]. This is because many vulnerabilities are not known during design time. Hence, it is always necessary to continue the investigation for finding novel threats that can arise in the system. In order to aid the security research to protect the confidentiality of additive manufacturing system, first of all we will present an analysis on the mathematical model of the fused deposition modeling based 3D printer. Where we, in detail, describe the components of the FDM based 3D printers, and provide various models and equations to enhance the knowledge about 3D printer systems, to design our novel attack model. Then we work on the leakage model, where we provide the side-channel model, vibration source analysis, acoustic leakage analysis, and leakage quantification to understand the relation between the cyber-data and acoustics. Then, finally we will present our novel acoustic side-channel attack model to breach the confidentiality of the system. It will consists of exploration of time and frequency domain features, learning algorithms trained to acquire specific information (axis of movement, speed of the nozzle) about the G-code, context-based post processing, and algorithms used to reconstruct the G-code by reverse engineering.

The rest of the thesis is organized as follows: Background and related work is presented in Chapter 2. Mathematical model of the system is described in Chapter 3. Leakage modeling and analysis are presented in Chapter 4. Attack model description is provided in

Chapter 5. Results are provided in Chapter 6. Challenges and future work are discussed in Chapter 7 before concluding this paper in Chapter 8.

CHAPTER 2: BACKGROUND & RELATED WORK

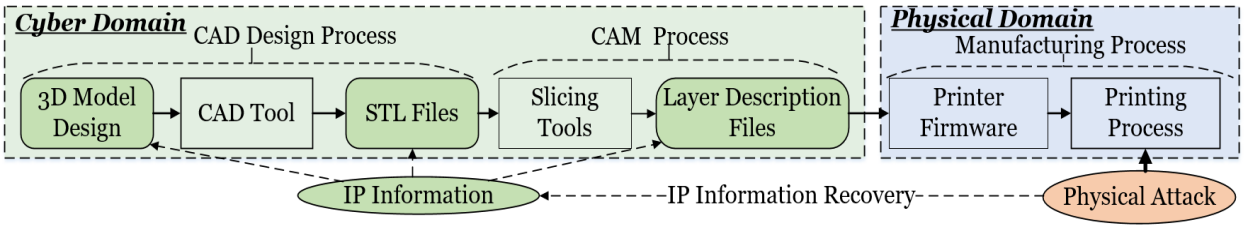


Figure 1: Confidentiality Breach during Printing Process of 3D Printers

A typical digital process chain in cyber-physical additive manufacturing systems is presented in Figure 1. Designers, first, start their design of 3D objects with 3D Computer-Aided Design (CAD) modeling tools such as Sketchup [28] and the extended version of Photoshop Adobe Photoshop CC [29]. Next, the CAD tool generates a standard STereoLithography (STL) for the manufacturing purpose. This STL file consists of description of triangulated surface, which make up the 3D model, using the unit normal and the vertices of the triangle [30]. The vertices of the triangles are expressed using three-dimensional Cartesian coordinate system. Computer Aided Manufacturing (CAM) process is then required to slice the STL file into layer-by-layer description file (e.g., G/M-code). Then, the layer description file is sent to the manufacturing system (e.g., 3D printer) for production. The 3D printer has a firmware which translates the G/M-codes into specific signals required to actuate the physical components of the system. G-code are responsible for the moving the various physical components, whereas the M-code are responsible for determining the machining parameters (such as temperature, coolant fan speed, etc.). In this thesis we focus our work on analyzing the G-code, as they describe the geometry of the 3D object, which encompasses the intellectual property in terms of the structural parameters of the 3D object.

In the physical domain of the additive manufacturing, components such as a stepper motor, fan, extruder, base plate etc., carry out operations on the basis of information provided by the cyber domain (G-code). In carrying out the operation, these physical components leak cyber domain information (G-code) from the side-channels, such as acoustic and power, which may be used to steal IP by performing physical-to-cyber domain attack. The issues regarding the theft of IP and the framework for preventing IP theft have been studied in [31] and [22]. The study of attack in the process chain, starting from the 3D object design to its creation, along with a case study of cyber-attacks in STL file, is presented in [32]. However, physical domain attacks are not well studied by the existing works. There are several publications utilizing the side-channel information to gather data related to the cyber domain in other systems. [33] has used the acoustics emanated from the dot matrix printer while printing to recover the text it was sent to print. Authors in [34] have been able to decode the keys pressed in the Enigma machine by analyzing the sound made by the device while pressing the keys. However, these methodologies are not applicable to 3D printers since, unlike printed words on paper, a 3D printer's movement has infinite possibilities. Recently, researchers from MIT have found that even the minor movement of physical devices can leak information about the cyber domain. In [35], they have successfully retrieved digital audio being played by capturing the vibration of objects near a sound source by a high speed camera. However, in a 3D printer, there are multiple sources of sound and vibration. Therefore, the task of analyzing sound for G-code reconstruction requires a completely new approach. Authors in [36] have considered using side-channel for providing security, but they have not demonstrated any methodology for using it to steal the IP. In

summary, the related work is focused on retrieving the text being printed (either in keyboard or dot matrix printer), analyzing acoustic emissions for observing mechanical degradation of the physical components in a manufacturing plant, etc. However, the possibility of using the acoustic emissions for reconstruction of a 3D object has not been considered. Hence, in this thesis, we have designed a novel acoustic side-channel attack model to breach the confidentiality of cyber-physical additive manufacturing systems.

CHAPTER 3: MATHEMATICAL MODEL OF THE SYSTEM

3.1 System Description

Additive manufacturing systems can be classified on the basis of the process and raw materials used [37]. State-of-the-art FDM based additive manufacturing systems consist of four to five two-phase stepper motors depending upon their structural design and number of filaments available for extrusion. Due to high torque/size ratio, and comparatively lower resonance and noise emission, hybrid stepper motors have been widely used in these 3D printers, and they are the main source of noise and vibration. Hence, our system model consists of four two-phase bi-polar hybrid stepper motors working in conjunction. Hybrid stepper motors use micro-stepping to make the transition of stator flux smooth. This in return reduces the vibration, and allows for smaller step angles [38]. The two-phase bi-polar hybrid stepper motors have eight poles for generating the stator flux. The fundamental law of energy conversion for stepper motors is shown in Figure 2. Here, the electrical energy (current $\boldsymbol{i}$) is first converted to electric and magnetic field ($\mathbf{F_{em}}$), which in turn guides the rotors. The electromagnetic field acting upon the various components produces force, which causes them to vibrate and produce sound ($\boldsymbol{P}$).

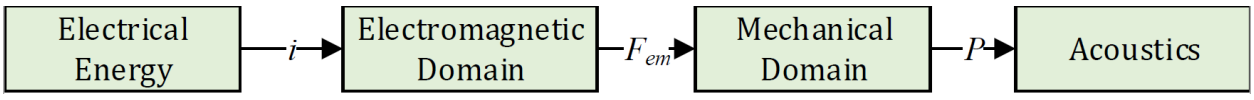


Figure 2: Energy Conversion in FDM based 3D Printer System

3.2 Electrical Model

The single winding circuit of a two-phase bi-polar hybrid stepper motor is shown in Figure 3. $\boldsymbol{v_A}$ is the terminal voltage, $\boldsymbol{R}$ is the stator winding resistance, $\boldsymbol{L_A}$ is the coil inductance, the mutual inductance between the coil of phase A and phase B is $\boldsymbol{M}$, and $\boldsymbol{e_A}$ is the back electromotive force (emf).

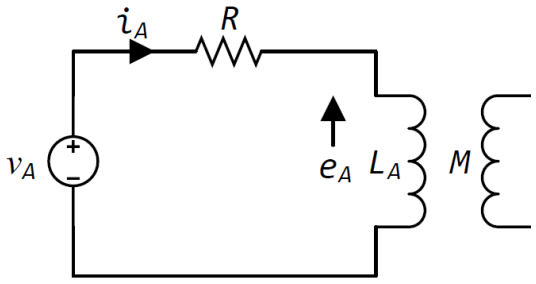


Figure 3: Electric Circuit of Phase A of a Stator Coil

The standard electrical model, as commonly found in many literature, e.g.,[39], for phase A of the hybrid stepper motor may be written as follows:

$$v_A = i_A R + L_A \frac{di_A}{dt} + M \frac{di_B}{dt} + e_A \tag{1}$$

For simplicity, lower case letters represent time varying variables, and capital letters denote time invariant variables. The permanent magnetic flux linkage $\boldsymbol{\Psi_A}$ is dependent on the mechanical rotational angle $\boldsymbol{\theta}$, and is given as follows:

$$\Psi_A = \Psi_m cos(p\theta) \tag{2}$$

Where $\boldsymbol{\Psi_m}$ is the maximum stator flux linkage, and p is the number of rotor pole pairs. Then the back emf induced in phase A may be written as follows:

$$e_A = \frac{d\Psi_A}{dt} = -p\Psi_m sin(p\theta)\frac{d\theta}{dt} \quad (3)$$

Similarly, the back emf induced in phase B of the stator winding may be written as follows:

$$e_A = \frac{d\Psi_B}{dt} = -p\Psi_m sin(p(\theta - \lambda))\frac{d(\theta-\lambda)}{dt} \quad (4)$$

Where $\boldsymbol{\lambda}$ is the angle between the two stator windings, i.e. phase A and phase B. From this electrical model, we can see that the time varying current passing through the stator core affects the magnetic field as well as the back emf produced in the circuit. The time varying current however is dependent on the supplied G-code to the 3D printer, which is in fact dependent on the structure of the 3D object. Hence, hence there is direct correlation between cyber-domain data and the current being supplied in the stepper motor.

3.3 Mechanical Model

The torque produced by current $\boldsymbol{i_A}$ for the given flux linkage $\boldsymbol{\Psi_m}$ may be written as follows [40]:

$$T_A = -p\Psi_m i_A sin(p\theta) \quad (5)$$

Similarly torque produced by current $\boldsymbol{i_B}$ may be written as follows:

$$T_B = -p\Psi_m i_B sin(p(\theta - \lambda)) \quad (6)$$

The Equations (5) and (6) explains that the torque production in stepper motor is dependent on the time varying current, which also happens to determine the magnetic flux linkage. This torque in turn determines the various kinds of vibration in the 3D printer, such as torque

ripple. However, the current is again determined by the cyber-domain data (G-code). Hence the system vibration in turn is dependent in the cyber-domain data.

3.4 Equation of Motion

The equation of motion for hybrid stepper based on Equations (3), (4), (5), and (6) is written as follows [40]:

$$J\frac{d^2\theta}{dt^2} + D\frac{d\theta}{dt} + p\Psi_m i_A \sin(p\theta) + p\Psi_m i_B \sin\big(p(\theta - \lambda)\big) = 0 \quad (7)$$

Where $\boldsymbol{J}$ is the moment of inertia of the rotor and the load combined, $\boldsymbol{J} = \boldsymbol{J_M} + \boldsymbol{J_L}$, and $\boldsymbol{D}$ is the damping coefficient based on eddy current, air friction, hysteresis effects, etc. From Equation (7), we can observe that the load attached to the stepper motor plays an important role in determining the system resonant frequency. This is an important observation, because even though same stepper motor might be used in the 3D printer to the printer nozzle in different axis, its resonant frequency varies when the load is different. Using Equation (7), we can determine the resonant frequency of the individual stepper motor in the 3D printer.

3.5 Equation of Radiated Sound

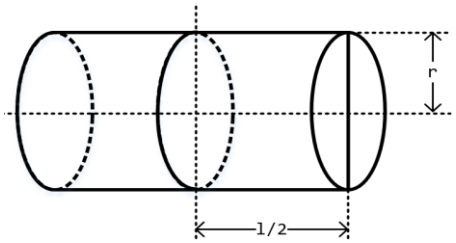


Figure 4: Outer Mechanical Structure of a Stepper Motor

For simplicity, we assume that the structure of the stator of the hybrid-stepper motor is cylindrical (as shown in Figure 4). With this, we may express the total sound power radiated by the electric machine due to the varying radial force acting upon the stator as follows [41]:

$$P = 4\rho c\pi^3 f^2 A_{rd}^2 rlI_{rel} \quad (8)$$

Where $\boldsymbol{P}$ is the radiated sound power (W), $\boldsymbol{\rho}$ is the density of the medium (kg/m^3), $\boldsymbol{c}$ is the speed of the sound in the medium (m/s), f is the excitation frequency of the vibration with multiple harmonics (Hz), $\boldsymbol{A_{rd}}$ is the surface vibratory displacement (m), $\boldsymbol{r}$ is the radius of the cylindrical stator (m), $\boldsymbol{l}$ is the length of the stepper motor (m), and $\boldsymbol{I_{rel}}$ is the relative sound intensity. $\boldsymbol{I_{rel}}$ depends on the mode of stator vibration $\boldsymbol{R}$, the radius, and the length-diameter ratio. Hence, stepper motors with different geometry and design in the 3D printer will emit different sound power. The attacker will have to consider this fact for designing successful attack models. Since the intensity of sound will degrade according to the inverse square law, placement and position of the audio recorder will affect the quality of sound acquired, and eventually the accuracy of the attack models. Equation (8) explains the power of radiated sound by a single stepper motor. However, in a 3D printer system, there are multiple stepper motors, each producing unique acoustic sound depending upon the cyber-domain data. Modeling this complex interaction is non-trivial. We have to first consider the various mediums through which the vibration will spread across the system, and analyze each source of vibration. To ease this task, in this thesis, based on the preliminary understanding of the mathematical model of the system based on the physical nature of stepper motors, we model the relation between the acoustic leakage from the system and the cyber-domain data, using a data-driven modeling approach.

CHAPTER 4: LEAKAGE MODELING & ANALYSIS

4.1 Side-Channel Model

Using an acoustic data acquisition device, an attacker may physically observe the vector $O_i = [o_1, o_2, o_3, \ldots, o_i]$. This is in fact the measurement of the acoustic power radiated by the stepper motor, which for a single stepper motor is given in Equation 8. Physically observable signal O corresponds to the side-channel leakage function L as follows:

$$O = L + N \tag{9}$$

Where N represents the Gaussian noise added to the side-channel leakage function L. Leakage function L depends on the G-code instruction G. For each G-code instruction G_i, the acoustic leakage vector $l_{(i,k)}$ acquired is of length k, and $g_i = [g_{(i,1)}, g_{(i,2)}, g_{(i,3)}, \ldots, g_{(i,k)}]$. The length k depends on the sampling frequency of the audio device used. The side-channel consists of two channels, where $G \rightarrow L$ channel leaks information about G in L, and $L \rightarrow O$ channel leaks information about L in O. O, L, and G are modeled as random variables on samples $o_{(i,k)}$, $l_{(i,k)}$, and g_i, respectively. The fundamental information contained in g_i are speeds $v_i=\{v_{(x,i)}, v_{(y,i)}, v_{(z,i)}\}$, axis movements $a_i=\{a_{(x,i)}, a_{(y,i)}, a_{(z,i)}\}$, positive or negative distance moved in each axis $d_i=\{d_{(x,i)}, d_{(y,i)}, d_{(z,i)}, d_{(e,i)}\}$, where $d_{(e,i)}$ is the extrusion amount measured in length of plastic deposited. Hence, the leakage function L is a function of all these parameters, given as follows:

$$o_i = l(f(v_i, a_i, d_i)) + n_i \tag{10}$$

Due to the efficiency of the profiled attacks, the estimation of leakage function L may be done using profiling acoustic traces O_i collected for parameters $\{v_i, a_i, d_i\}$during the training phase.

We argue that any G-code instruction g_i can be broken down to its corresponding parameters, such that g_i={v_i, a_i, d_i}. Moreover, a data-driven modeling approach using machine learning algorithms will be used to estimate the mapping function $g_i = \hat{f}(o_i, \alpha_n)$. Where α_n is the tuning parameter for the function. Due to the presence of multiple parameters, classification and regression machine learning algorithms will be used to estimate the functions. In the consequent sections, we will describe the relation between the various parameters {v_i, a_i, d_i } and the equations described in the Chapter 3. This will allow us to determine specific machine learning algorithms to estimate the corresponding parameters from the leakage signal, and design the overall attack model.

4.2 Natural Rotor Oscillation Frequency

The radiated sound power is higher when the stepper motor vibrates with the rotor's natural oscillation frequency. Using Equations 1, 3, 4, and 7, we may calculate the natural frequency of rotor oscillation as follows [39]:

$$\omega_{np}^2 = \frac{2p^2 \Psi_m I_o cos(\frac{p\lambda}{2})}{J} \tag{11}$$

Where I_o is the stationary current flowing in the two phases A and B. When the stepper motor is rotating with the harmonic frequency of the natural frequency such as …,$\frac{\omega_{np}}{4}$, $\frac{\omega_{np}}{3}$, $\frac{\omega_{np}}{2}$, $2\omega_{np}$, $3\omega_{np}$, $4\omega_{np}$,… the vibration is more prominent due to resonance. Equation (11), describes the natural frequency of a single stepper motor when it is not attached to any mechanical structure. In a complex system such as 3D printer, the natural rotor oscillation frequency will vary according to the amount and type of load handled by each of the stepper motor.

4.3 Stator Natural Frequency

The natural frequency of the stator depends on the vibration modes. Due to the prominence of the radial force acting on the stator, we will consider only the circumferential radial vibration modes and the corresponding stator natural frequencies. The structure of the stator is complex and many attempts have been made to calculate the natural frequencies of the stator with various considerations, an example being single-ring type stator [42][43]. Since the external structure connected to the stator also influences its mass and stiffness, the natural frequency of the stator with circumferential vibration mode m and axial vibration mode $\boldsymbol{n}$ of the frame may be calculated as follows [44]:

$$\omega_{stator\ np}^2 \approx \frac{K_m^{(c)} + K_{mn}^{(f)}}{M_c + M_f} \tag{12}$$

Where $\boldsymbol{K_m^{(c)}}$ is the lumped stiffness of the stator core, $\boldsymbol{K_{mn}^{(f)}}$ is the lumped stiffness of the frame, and $\boldsymbol{M_c}$ and $\boldsymbol{M_f}$ are the mass of the stator and the frame, respectively. Equation 12 has been derived by assuming that the lumped stiffness of the core and the frame are in parallel. Equation 12 states that the stator natural frequency depends on the frame structure to which the stepper motor is connected.

4.4 Source of Vibration

The main sources of vibration in stepper motors are electromagnetic, mechanical, and aerodynamic [41]. These vibrations help in radiating sound from the stepper motor stator surface and the frame to which the motor is connected. In this thesis, we will consider the electromagnetic and mechanical sources as they are the major sources of leakage.

1.1.1 Electromagnetic Source

The fundamental source of vibration in hybrid stepper motors is due to the fluctuation of electromagnetic force produced by the winding of the stator. The two types of vibration produced by the electromagnetic force are:

i) Radial Stator Vibration: In a hybrid stepper motor, both stator and the rotor are responsible for exciting the magnetic flux density in the air gap between the rotor and the stator. These magnetic flux contribute in generating the radial force. If $\boldsymbol{\sigma_{(l,k)}}$ be the radial force at pole $\boldsymbol{l}$ for $\boldsymbol{k^{th}}$ harmonic then the total radial force acting on the stepper motor may be calculated as follows [45]:

$$\sigma_{total} = \sum_{k=1}^{\infty} \sigma_{(l,k)} cos(k\omega t + \varphi_{(l,k)}) \tag{13}$$

Where $\boldsymbol{\varphi_{(l,k)}}$ is the phase angle of the radial force at pole $\boldsymbol{l}$ for $\boldsymbol{k^{th}}$ harmonic, $\boldsymbol{\omega = 2\pi f}$, and $\boldsymbol{f}$ is the frequency determined by the stepping rate of the motor. This radial force acts on the stator and rotor surface and deforms its structure. This produces vibration and eventually sound in the stepper motor. When the radial force excites the harmonics of the natural frequencies of the stator/frame structure and the rotor oscillation, vibration is more prominent due to resonance.

ii) Torque Ripple: Even though torque ripple is substantially reduced by using the micro-stepping for driving the stator windings, micro-stepping position ripple is still produced due to non-conformity to the ideal sine/cosine waves required for absolute removal of the torque ripple. However, the vibration produced by the torque ripple is less compared to the radial stator vibration.

4.4.1 Mechanical Source

The rotor and load connected to the stepper motor may also produce vibration and sound at various frequencies due to friction, rotor unbalance, shaft misalignment, loose stator laminations, etc. These vibrations produce a loud noise due to resonance.

4.5 Acoustic Leakage Analysis

We have so far modeled the source of vibration and noise in hybrid stepper motors. In this section, we will analyze the leakage channel $\boldsymbol{G \rightarrow L}$ to demonstrate the relation between the G-code parameters $\{\boldsymbol{v_i}, \boldsymbol{a_i}, \boldsymbol{d_i}\}$ and the acoustic leakage.

LEMMA 1. *Given the acoustic leakage L in the channel* $\boldsymbol{G \rightarrow L}$*, the frequency of the radiated sound varies according to the speed of the nozzle in X and Y axis, respectively.*

PROOF: The radial force generated in Equation 13 in each pole depends on the magnetic flux density, stator tooth width, and rotor cap thickness. The magnetic flux density depends on the current passing through the each winding. To increase the angular speed of the stepper motor, the stepping rate is increased. From Equation 13, we can see that this increases the frequency of the radial force acting on the stepper motor. From Equation 8, we also can see that the radiated power increases with the excitation frequency of the vibration.

LEMMA 2. *Given the acoustic leakage L in the channel* $\boldsymbol{G \rightarrow L}$*, the power frequency spectrum of the radiated sound from the stepper motors X, Y, Z, and the one for the extruder are different.*

PROOF: The natural rotor oscillation frequency in Equation 11 is inversely proportional to the moment of inertia of the load and the motor ($J_L + J_M$). The load moved by each stepper motors X, Y, Z, and E are different in state-of-the-art stepper motors. The natural frequency in Equation 12 also depends on the mechanical structure of the frame to which the stepper motor is connected. Due to the mechanical structure of the 3D printers, stepper motors are placed in various locations and are connected to different frame structures. Therefore, the natural frequencies of the stepper motors vary according to the load and the frame to which they are attached. This means that the resonance can occur at different frequencies of the vibration for different stepper motors and the frame structure. This causes the power spectrum of the radiated sound to vary according to the source of sound, i.e., the stator motor and the frame structure.

LEMMA 3. *Given the acoustic leakage L in the channel* $G \rightarrow L$*, the intensity of the radiated power will vary according to the direction of the nozzle movement in different directions from the audio device.*

PROOF. According to the inverse square law, the intensity of the sound decreases drastically with the square of the distance from the sound source. If P is the power of the sound source and r be the distance from the sound source, then we have:

$$I = \frac{P}{4\pi r^2} \tag{14}$$

Hence, for analyzing the direction of movement, the intensity of the sound radiated by each motor and frame structure may be measured.

4.6 Success Rate Calculation

In our attack model, the reconstruction of the G-code depends on the separability of the different parameters such as axis of movement of the nozzle, i.e. X, Y , and Z. Apart from this, the capability of the attacker to predict the speed of the nozzle movement ($\boldsymbol{v_x, v_y, v_z}$) in each of the axes, will determine the success rate of the attack model. Hence, success can simply be quantified by measuring the separability of the nozzle movement and prediction accuracy of the nozzle speeds in each axis. The separability is specifically measured using the Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve, whereas the speed prediction accuracy can be measured using the Mean Square Error (MSE). We also calculate the accuracy of the classifiers as follows:

$$Accuracy = \frac{TP+TN}{Total\ Sample} \tag{15}$$

Where True Negative (***TN***) and True Negative (***TN***) are the total number of right classifications made by the classifiers.

CHAPTER 5: ATTACK MODEL DESCRIPTION

5.1 Attack Model

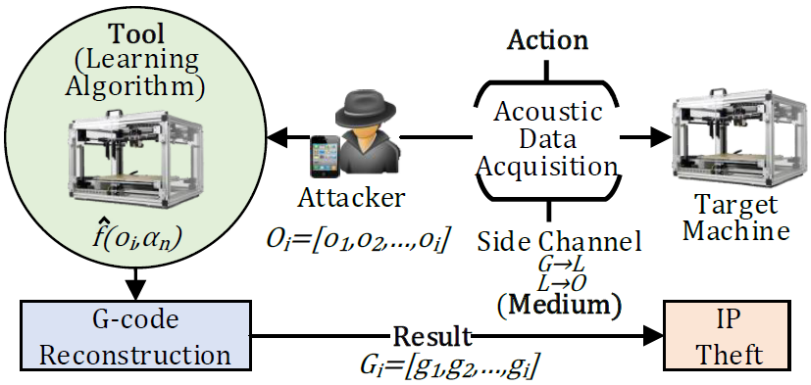


Figure 5: Acoustic Side-Channel Attack Model

In our attack model (shown in Figure 5), we implemented a novel way of leveraging the stochastic dependency of the acoustic leakage $\boldsymbol{O}$ to $\boldsymbol{L}$ in channel $\boldsymbol{G \rightarrow L}$ and eventually dependency of $\boldsymbol{L}$ to $\boldsymbol{G}$ in channel $\boldsymbol{L \rightarrow O}$. However, in order to extract the information about $\boldsymbol{G}$, we developed multiple classification and regression models using known machine learning algorithms to estimate the parameters $\{\boldsymbol{v_i}, \boldsymbol{a_i}, \boldsymbol{d_i}\}$. Instead of estimating single function $\boldsymbol{g_i = \hat{\hat{f}}(o_i, \alpha_n)}$, we further break it down to the problem of estimating the functions $\boldsymbol{a_i = \hat{\hat{f}}(o_i, \alpha_n)}$, $\boldsymbol{v_i = \hat{\hat{f}}(o_i, \alpha_n)}$, and $\boldsymbol{d_i = \hat{\hat{f}}(o_i, \alpha_n).}$ We have $\boldsymbol{a_i}=\{\boldsymbol{a_{(x,i)}}, \boldsymbol{a_{(y,i)}}, \boldsymbol{a_{(z,i)}}\}$, as random variables denoting the presence of the nozzle movement in X,Y, and Z axis, respectively. Where $\{\boldsymbol{a_{(x,i)}}, \boldsymbol{a_{(y,i)}}, \boldsymbol{a_{(z,i)}}\} \in \{0,1\}^B$, where binary value 0 represents the absence of nozzle movement and 1 represents the presence of the nozzle movement. We have $\boldsymbol{v_i}=\{\boldsymbol{v_{(x,i)}}, \boldsymbol{v_{(y,i)}}, \boldsymbol{v_{(z,i)}}\} \in \mathbb{R}_+$, and $\{\boldsymbol{d_{(x,i)}}, \boldsymbol{d_{(y,i)}}, \boldsymbol{d_{(z,i)}}\} \in \mathbb{R}$. The attack consists of two phases. In the first phase, a training cyber-data (G-code) is designed to collect range of acoustic emissions from a device which is same

or at least similar to the target device. Based on the data collected, various functions are estimated using the data-driven modeling approach. Then during the second phase, the attack phase, attackers collects the acoustic emissions when a real 3D object is being printed by the target device. Using the estimated functions, various parameters of the G-code are predicted, and then combined to reverse-engineer the full G-code, and hence the 3D object, effectively stealing the intellectual property hidden in the geometry of the object.

5.2 Leakage Exploitation

The accuracy with which an adversary is able to exploit the acoustic leakage depends on their ability to estimate the functions $\hat{\dot{f}}(.)$. Breaking the process for estimating $g_i = \hat{\dot{f}}(o_i, \alpha_n)$ into multiple estimation functions improves the adversarial attack model by focusing on only those parameters in G that are required for breaching the confidentiality of the system. However, the accuracy of the attack model now becomes a function of successful estimation of individual functions.

LEMMA 4. *The observable leakage vector* $O_j = [o_1, o_2, ..., o_j]$ *sampled from the channel* $L \rightarrow O$ *correspond to* $G_i = [g_{(i,1)}, g_{(i,2)}, ..., g_{(i,k)}]$ *such that* $k = \Delta t \times f$ *and* $j \div k = i$*, where* Δt *is the length in time* G_i *leaks analog emissions in the acoustic side-channel.*

PROOF: The Sequence of G-codes supplied to the 3D printer is discrete. However, the duration of the sound power radiated by the printer for the corresponding G-code varies. Hence, the leakage O observed in channel $L \rightarrow O$ corresponds to the duration of each of the G-code instructions and the sampling frequency of the data acquisition device. Hence, for

each k length for G_i, we have $[o_{(i,1)}, o_{(i,2)}, \ldots, o_{(i,k)}]$ as the vector of observable physical emissions.

ASSUMPTION 1. *The direction of d_z during the printing is always in either positive or negative Z-axis.*

In additive manufacturing systems, materials are extruded layer wise. Hence, direction in Z-axis should always be in one direction. This allows us to exclude the estimation of direction motor in Z-axis.

ASSUMPTION 2. For the given vectors G_i, we have acoustic observable traces $[o_{(i,1)}, o_{(i,2)}, \ldots, o_{(i,k)}]$, and all the observable traces with length k are similar.

Since a single G-code instruction actuates similar physical behavior for certain amount of time (e.g., moving the nozzle in X-axis with 500 mm/min speed for 10 mm) the acoustic emission will be similar.

5.3 Components of the Attack Model

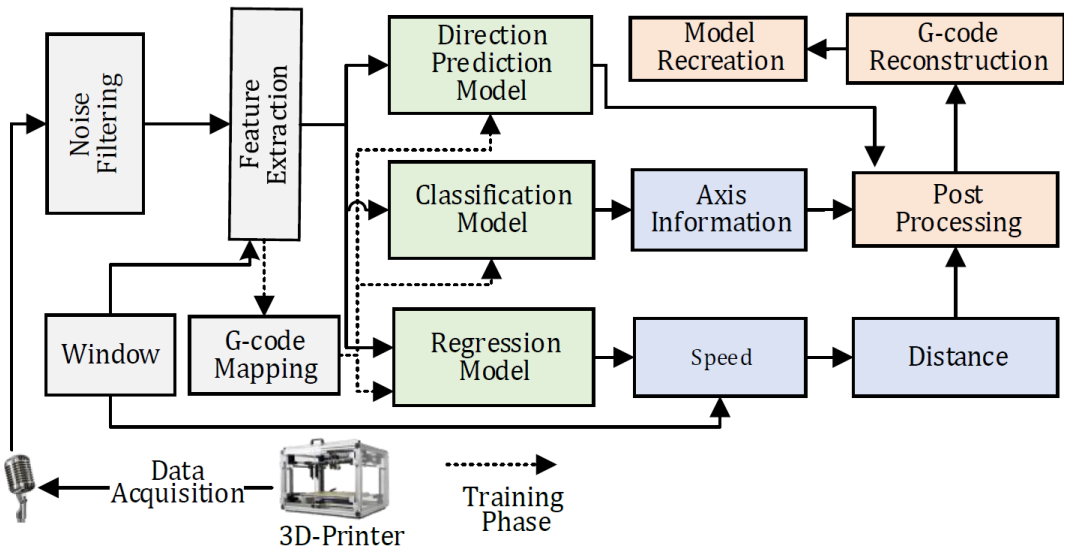


Figure 6: Components of the Attack Model

Components of the acoustic attack model is presented in Figure 6. As explained in Section 5.1, the attack process consists of two phases, training and the attack phase. We will now go through each of the components of the attack model.

5.3.1 Data Acquisition

The first step in acquiring the observable leakage trace, $\boldsymbol{Oi=[o_1, o_2, ..., o_i],}$ is to place an audio recording device such as a mobile phone near the 3D printer. The sampling frequency of the recording device must be higher than 40 kHz to capture the sound in the audible range to avoid aliasing effect [46]. The distance of the audio device from the 3D printer and the angle to the different sources of sound (stepper motor X and stepper motor Y) will also determine the accuracy of leakage exploitation. Hence, brute force may be used to find the optimal position of the recorder to acquire the best possible acoustic analog emissions. However, exploration of the position of the audio recorder for optimal emission acquisition is beyond the scope of this thesis.

5.3.2 Noise Filtering

We use a digital finite impulse response band pass filter to eliminate the noise from low frequency alternating current from the power source, and the high frequency noise generated by the hybrid stepper motor winding when it is in charged and in idle state. The passband frequency for the noise removal is between 100 Hz and 20 kHz.

5.3.3 Feature Extraction

We use features commonly used in speech pattern recognition [47] in the time and frequency domains to train our learning algorithms. In the time domain, the features extracted are *frame energy, Zero Crossing Rate (ZCR), energy entropy*. The features extracted from the

frequency domain are *spectral entropy, spectral flux, Mel-Frequency Cepstral Coefficients* (MFFC), and energy of *Short-Time Fourier Transform* (STFT) divided linearly into frequency intervals. The features are extracted from a set of frames with fixed frame size of 50ms. However, better timing resolution can be obtained by making the frame size smaller, and for frequency resolution, larger frame size is required. Thus, the size of the frame is subjective to the type of features being extracted, and exploration of the window size is beyond the scope of this thesis. From each frame, we extract features and create a feature vector to supply the training algorithm. For a given frame of length $\boldsymbol{F_L}$ with audio signals $\boldsymbol{x(i) = 1, 2 \ldots F_L}$, different features are extracted as follows:

$$Frame\ Energy\ (E) = \sum_{i=1}^{F_L} |x(i)^2| \tag{16}$$

Frame energy is enough to predict direction when the printer is only printing in one axis, however *spectral energy* is required while predicting the direction in multiple axes movement. *ZCR* is calculated as follows:

$$ZCR = \frac{1}{2F_L} \sum_{i=2}^{F_L} |sign[x(i)] - sign[x(i-1)]|^2 \tag{17}$$

ZCR is high when the printer is not making any sound, due to the noise, and low when it is printing. For *energy entropy*, we divide the frame into short frames of length $\boldsymbol{K}$. If $\boldsymbol{E_j}$ is the energy of the $\boldsymbol{j^{th}}$ short frame, then we have:

$$Energy\ Entropy = -\sum_{j=1}^{K} e_j log_2(e_j) \tag{18}$$

$$where\ e_j = \frac{E_j}{\sum_{i=1}^{K} E_i} \tag{19}$$

Energy entropy measures the abrupt change in the energy of the signal, and may be used to detect the change of motion. For frequency domain data, let $\boldsymbol{X_i(k),\ k = 1,\ 2 \ldots F_L}$ be the magnitude of the Fast Fourier Transform (FFT) coefficient of the given frame. *For spectral entropy*, we divide the spectrum into $\boldsymbol{L}$ sub bands. Let $\boldsymbol{E_f}$ be the energy of the $\boldsymbol{f^{th}}$ sub band then we have:

$$Spectral\ Entropy = -\sum_{f=1}^{L-1} n_f log_2(n_f) \tag{20}$$

$$where\ n_f = \frac{E_f}{\sum_{j=1}^{L-1} E_j} \tag{21}$$

Spectral flux measures spectral change between two successive frames, and can be used to detect the change of speed of the nozzle while printing within each layer.

$$Spectral\ Flux_{i,i-1} = \sum_{k=1}^{F_L} (EN_i(k) - EN_{i-1}(k))^2 \tag{22}$$

$$where\ EN_i(k) = \frac{X_i(k)}{\sum_{j=1}^{F_L - 1} X_i(j)} \tag{23}$$

To gather more information from the spectral data, we have calculated *cepstrum* (inverse FFT of the log magnitude of the FFT of a signal) features. *MFFC*, which uses non-linear separation of frequency intervals (perceived as equally spaced by human ears), are more efficient in audible sound pattern recognition [48], hence we have incorporated *MFFC* in our feature vector. We have also computed features by linearly dividing the *STFT* of the audio signals in an interval of 100 Hz, and placing the average of the energy in this interval in one bin. These features are more efficient to implement than the *MFFC* while performing spectral subtraction when two motors are running simultaneously.

5.3.4 Regression Model

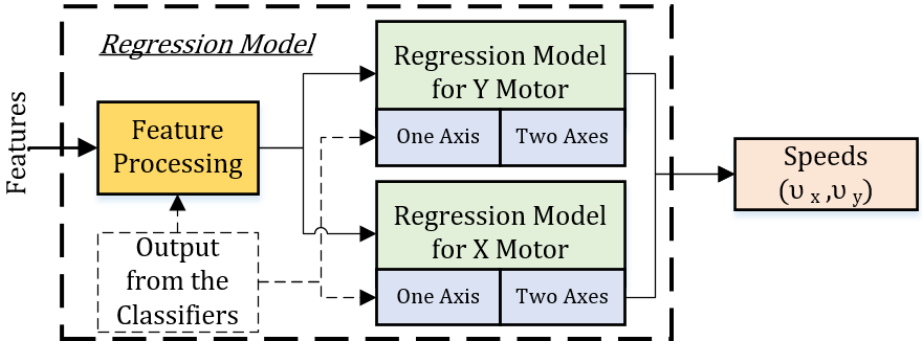


Figure 7: Regression Model for Nozzle Speed Prediction in X and Y Axis

The regression model consists of a collection of models, each using a supervised learning algorithm for regression as shown in Figure 7. These models are used for estimating the functions $v_{(x,i)} = \hat{f}(o_i, \alpha_n)$ and $v_{(y,i)} = \hat{f}(o_i, \alpha_n)$. These functions are used to extract information about speed in X direction given only one axis movement, and speed in X direction given the motion in two axis. Similarly, this is done for speed in Y direction as well. Hence, we have four regression model to predict the nozzle speed in XY-plane.

ASSUMPTION 3. *The speed in the Z direction while printing the given model with the given printer is fixed and the speed of extrusion can be calculated as a function of layer height and nozzle diameter.*

For a given 3D printer, the layer height is assumed to be fixed. This relaxes the complexity for leakage exploitation by reducing the need for estimating speeds v_x and v_e. The speed of the printing, also known as the travel feed-rate is determined by training these regression algorithms [49]. After gaining the information about the travel feed-rate, we may calculate the distance moved by the nozzle as follows:

$$Distance = Framesize \times Speed \tag{24}$$

Where ***Framesize*** is measure in millisecond, and the ***Speed*** is measured in millimeter per millisecond. When the nozzle is moving in only one axis, the regression model may just take the features directly without further processing, however, when the nozzle is moving in two or more axes, the audio signal from one motor is combined with the others. Hence, it becomes imperative to separate these signals before the regression model can be used to predict the speed.

ALGORITHM 1: Feature Processing and Speed Calculation with Motion in XY Axes.

Input: Feature Vectors $xy_\beta, x_\beta, y_\beta$ // $\beta \rightarrow$ Total features
Output: Speed ϑ_x, ϑ_y // $\vartheta \rightarrow$ Speed

1 $\vartheta^i_{x_{mean_{\beta j}}} = \frac{1}{N_i}\sum_{n=1}^{N_i} x_{\beta_{n,j}}$ // $j = 1 \longrightarrow \beta n$ $N_i \longrightarrow Total\ data\ for\ speed\ i$
2 $\vartheta^i_{y_{mean_{\beta j}}} = \frac{1}{N_i}\sum_{n=1}^{N_i} y_{\beta_{n,j}}$
3 **for** *each xy* **do**
4 **for** ϑ^i *in* $range(v^1, v^n)$ // n:Total number of speed used in training
5 **do**
6 $xy_{beta(xy-y)} = xy_{beta} - \vartheta^i_{y_{mean_\beta}}$
7 $\vartheta^i_x \leftarrow RegressionModel1(xy_{beta(xy-y)})$
8 $xy_{beta(xy-x)} = xy_{beta} - \vartheta^i_{x_{mean_\beta}}$
9 $\vartheta^i_y \leftarrow RegressionModel2(xy_{beta(xy-x)})$
10 $diff_i = \mid \vartheta^i_y - \vartheta^i \mid$
11 $\vartheta_y = \vartheta^i_y with\ minimum\ diff_i$
12 $\vartheta_x = \vartheta^i_x with\ minimum\ diff_i$
13 **return** ϑ_x, ϑ_y

Algorithm 1 provides the pseudo code for performing the spectral subtraction necessary when motion is involved in both the X and Y axes. It takes features, extracted from the audio when both the X and Y motors are running, and the features from the training phase for individual motor X and Y as the input. Spectral subtraction is not performed for Z motor because it only moves one layer at a time and the distance it moves is normally fixed for a given object. While training, n number of speeds, in incremental number is taken to train the regression models. For each of these speeds, lines 1 and 2 calculate the average magnitude

of spectral features. Then, for each of the speeds, line 6 assumes the speed of the Y motor and the spectral components are subtracted from the combined spectral features of X and Y. By subtraction, we remove the spectral components present in Y from the combination of these features. Line 7 gives the predicted speed for the given value of speed in Y direction. We use this speed to subtract the spectral features of X in the particular speed and again use this value to predict the speed for motion in Y-axis. In lines 11 and 12, the speed of X and Y that gives the minimum difference in the predicted speed and output speed in Y-axis is chosen as an output.

5.3.5 Classification Model

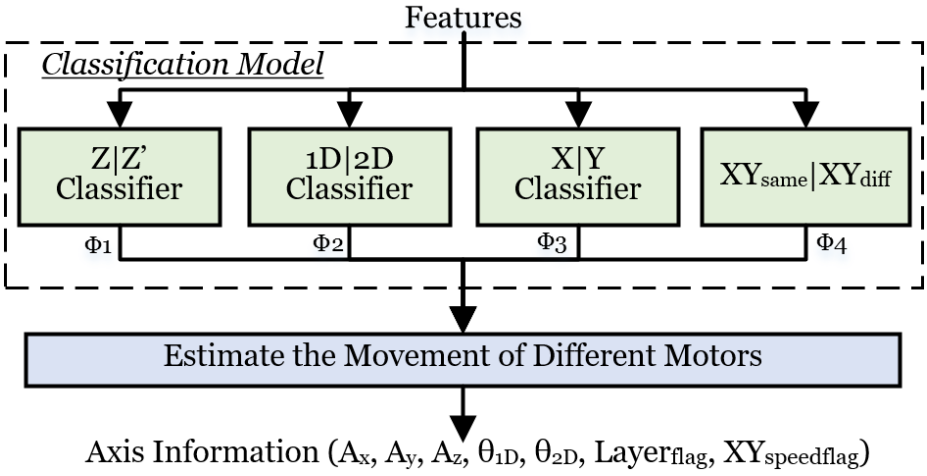


Figure 8: Classification Model for Axis Prediction

As shown in Figure 8, to determine the axis in which the nozzle is moving, the classification model consists of collection of classifiers to convert the classification problem into two-class separation model. On top level, this in fact will estimate the functions $\boldsymbol{a_{(x,i)} = \dot{\hat{f}}(o_i, \alpha_n)}$, $\boldsymbol{a_{(y,i)} = \dot{\hat{f}}(o_i, \alpha_n)}$, and $\boldsymbol{a_{(z,i)} = \dot{\hat{f}}(o_i, \alpha_n)}$. However, we have four classifiers that classify whether the Z-axis movement is present or not, if the nozzle movement is in single axis (just X or Y Axis) or

two axis (both in X and Y Axis), if the nozzle movement is just in X axis or just Y axis, and finally, when the movement is in both X and Y axis, if they have same speed or different speed in each of the axes. We have found that this model gives us better prediction results than multi-class classifier models. Each of these classifiers consists of supervised learning algorithms for classification.

ALGORITHM 2: Estimate the Axis of Movement.

Input: Classifier Outputs $\phi_1, \phi_2, \phi_3, \phi_4$
Output: Axis Parameters $A_x, A_y, A_z, \Theta_{1D}, \Theta_{2D}, Layer_{flag}, XY_{speedflag}$

```
Θ1D = 0, Θ2D = 0                                   // A ⟶ axis, Θ ⟶ dimension
Layer_flag = 0, XY_speedflag = 0
A_x = 0, A_y = 0, A_z = 0                          // Initialize to zero
if φ1 == 1 then
    Layer_flag = 1, A_z = 1
else
    if φ2 == 1                                     // One dimension movement
     then
        Θ1D = 1
        if φ3 == 1                                 // Movement in X-axis
         then
            A_x = 1
        else
            A_y = 1                                // Movement in Y-axis
    else
        Θ2D = 1, A_x = 1, A_y = 1
        if φ4 == 1                                 // X and Y move with same speed
         then
            XY_speedflag = 1
        else
            XY_speedflag = 0                       // Different speed
return A_x, A_y, A_z, Θ1D, Θ2D, Layer_flag, XY_speedflag
```

Algorithm 2 gives the pseudo code which takes the output from the classifiers to determine the axis of movement. It also gives information such as whether the layer has changed or not, and whether the nozzle is moving in X and Y axis with the same or different speed.

5.3.6 Direction Prediction Model

Most of the 3D printers have motors in a fixed location. However, the base plate, the nozzle or combination of both are always in motion while printing. Therefore, vibration is

conducted from the motor to the nozzle and the base plate of the printer. This means that the audio source physically gets closer or away from the recording device while printing.

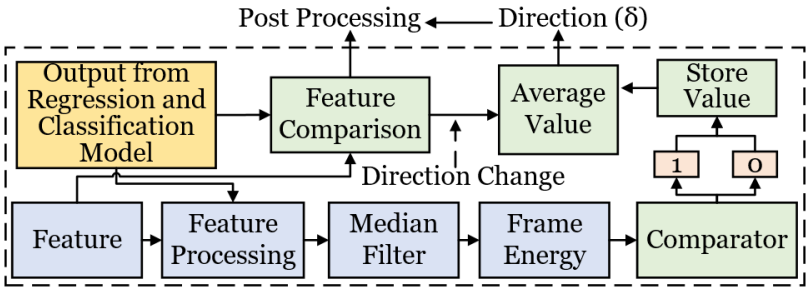


Figure 9: Direction Prediction Model

As shown in Figure 9, we can use the frame energy of the audio signal to check the direction of motion. For multiple motor movements, we utilize the difference of feature in frequency domain to calculate the energy of only those spectral components that represent the specific motor. In order to suppress the high fluctuation, median filtering is applied to the sequence of frame energies to smooth the curve of frame energies. The prediction model will output 1 if the frame energy is increasing and 0 if the frame energy is decreasing. In order to aid the direction prediction model and the post-processing, a feature comparison block measures the distance (Euclidean distance) between consecutive frame features. If the motion of direction changes, then there is a large difference in the features between the consecutive frames. We use this spike to detect the change in direction of motion of the nozzle.

5.3.7 Model Reconstruction

For reconstructing the G-code, we need to determine whether the 3D printer nozzle is actually extruding the filament or not. From our analysis, we have found that the printer nozzle moves at a higher speed when it is not extruding the filament. Hence, determining whether it is printing or not printing becomes a task of finding out the speed at which the

nozzle is moving. This information is acquired from the regression model. The extrusion amount for a given segment is machine-specific, and can be calculated as a function of the layer height, and the nozzle diameter. After acquiring the output from the regression model, classification model, and direction prediction model, Algorithm 3 calculates the positive or negative distance movement.

ALGORITHM 3: Calculate Distance Moved in Each Axis, and Check Extrusion.

Input: Output from Classifier and Regression Models $\vartheta_x, \vartheta_y, \vartheta_z$,w, $A_x, A_y, A_z, \delta_x, \delta_y, \delta_z$
Output: Distance Values d_x, d_y, d_z, d_E // $d \rightarrow Distance, x_\vartheta \rightarrow Speed\ in\ X\text{-}axis$

1 $d_x = 0, d_y = 0, d_z = 0$ // $w \rightarrow Frame\ length, A \rightarrow AxisFlag, \delta \rightarrow Direction$
2 **for** *each* i *in* x, y, z **do**
3 **if** $A_i == 1$ // Axis flag set
4 **then**
5 **if** $\delta_i == 1$ **then**
6 $d_i = \vartheta_i \times w$ // Positive distance
7 **else**
8 $d_i = \neg\vartheta_i \times w$ // Negative distance
9 **if** $\vartheta_x >= Speed_{High}$ || $\vartheta_y >= Speed_{High}$ **then**
10 $d_E = 0$ // $No\ extrusion\ in\ high\ speed$
11 **else**
12 $d_E = e_d$ // $e_d \longrightarrow Machine\ specific\ extrusion$
13 **return** d_x, d_y, d_z, d_E

Finally, Algorithm 4 reconstructs the G code for the printed object. It takes the input such as the distance moved in each of the axis, extrusion length, and the window size, and calculates the travel feed-rate (line 2).

ALGORITHM 4: Generate G-code of the Object.

Input: Distance and Frame Length d_x, d_y, d_z, d_E, w
Output: *G-code*

1 $dr_x = 0, dr_y = 0, dr_z = 0, dr_E = 0$ // Initialize to zero
2 $\vartheta = \frac{\sqrt{d_x^2+d_y^2+d_z^2}}{w}$ // Travel feedrate
3 $dr_x = dr_x + d_x$ // Distance moved in X-axis
4 $dr_y = dr_y + d_y$ // Distance moved in Y-axis
5 $dr_z = dr_z + d_z$ // Distance moved in Z-axis
6 $dr_E = dr_E + d_E$ // Extrusion amount
7 $G\text{-}code \leftarrow G1\ F(\vartheta)\ X(dr_x)\ Y(dr_y)\ Z(dr_z)\ E(dr_E)$
8 **return** *G-code*

And finally combines all the information (line 7) to construct each line of G-code necessary to reconstruct the object.

5.3.8 Post-Processing for Model Reconstruction

We have found a high mutual information between the G-code and the sound retrieved from the physical medium. For G-codes, let ***G*** be a discrete random variable with ***f(g)*** as its probability distribution function at ***g***. Let ***O*** be a discrete random variable representing the feature extracted from the acoustics with ***f(o)*** as its probability distribution function. Then the entropy of each of these random variables may be given as:

$$H(G) = -\sum_{g \in G} f(g)\, log_2\, f(g) \tag{25}$$

$$H(O) = -\sum_{o \in O} f(o)\, log_2\, f(o) \tag{26}$$

If ***f(g,o)*** and ***f(g|o)*** are the joint and conditional probabilities of the random variables, respectively, then the conditional entropy ***H(G|O)*** is calculated as:

$$H(G) = -\sum_{o \in O} \sum_{g \in G} f(g, o)\, log_2\, f(g|o) \tag{27}$$

The conditional entropy measures the amount of information required to describe outcome of a random variable G, given the information about a random variable O. In this context, in addition to the information gathered from O, the amount of additional additive manufacturing context-based information required to reconstruct the G-code is directly related to the mutual information. This is calculated as:

$$I(G; O) = H(G) - H(G|O) \tag{28}$$

We have found that the uncertainty of reconstruction of G-code or the entropy ***H(G|O)*** increases when the distance of the microphone is further away from the printer or when

there is added noise in the environment. It also increases when the speed of the printer is high and there are more short and rapid movements. During these scenarios, we can use the properties of additive manufacturing to post-process the data achieved from the learning algorithms. Specifically, we have used two post-processing stages which utilizes specific additive manufacturing context-based information.

Post-Processing Stage I: In this stage, we reduce ***H(G|O)*** by utilizing the fact that until the change of motion occurs, the nozzle moving in one particular dimension with a particular speed has a similar feature vector. By taking the output from the feature comparison model, we segment the acquired acoustic data into sections with similar movement. In this post-processing stage, we then choose the output of the classifiers to be the highest occurring value in the given segment, and for regression, we average the speed obtained within the same section. This is similar to averaging used in digital signal processing to increase the Signal-to-Noise Ratio (SNR).

$$SNR_{dB} = 10log_{10}(\frac{Power_{signal}}{Power_{Noise}}) \tag{29}$$

When we increase the SNR, the entropy of the signal is reduced. As there is high correlation among the features extracted from successive frames of the audio collected from the 3D printer, averaging the output of the classification and the regression model increases the SNR and thus reduces ***H(G|O)***.

Post-Processing Stage II: After applying post-processing stage I, the second stage measures the similarity between the two layers. The similarity of two layers is measured in terms of

number of segments, the sequence of motions in each layers, and the length of each segment. This post-processing stage helps the attack model in reducing the error due to miscalculated direction and fluctuating lengths by taking the average of segment lengths and direction among the similar layers of the 3D object.

5.4 Attack Model Training and Evaluation

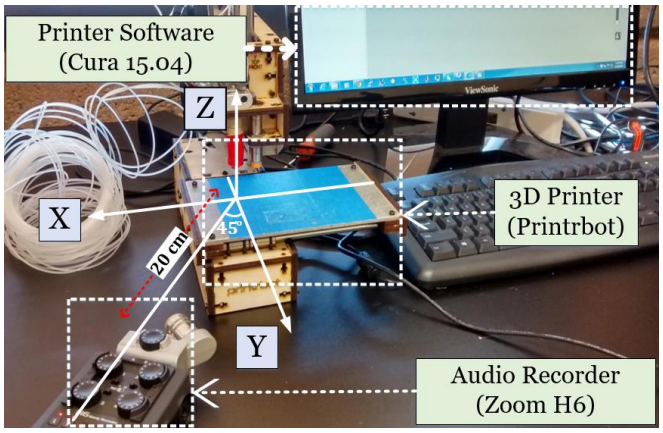


Figure 10: Experimental Setup for Training and Testing the Attack Model

Our testbed, shown in Figure 10, consists of a Printrbot 3D printer [50] with open source marlin firmware. It has four stepper motors. Motion in the X-axis is achieved by moving the base plate, whereas the nozzle itself can be moved in the Y and Z directions. The audio is recorded using a cardiod condenser microphone (Zoom H6) [51], which has a sampling frequency of 96 kHz and stores the data at 24 bit per sample. We have placed the audio recorder within 20 cm of the 3D printer. From our experiments, we have analyzed that for the direction prediction model to work efficiently, the audio device has to be placed at 45^{o} angle to both the X and the Y-axis as shown in Figure 10. This allows the audio device to capture the variation of sound in both X and Y directions. The digital signal processing,

feature extraction, and post-processing are performed in MATLAB [52], whereas the training of learning algorithms, their evaluation and testing is done using Python [49]. The attack model consists of supervised learning algorithms. For training these algorithms, initial training data has to be determined. The training data consists of G-code to move the printer nozzle at different speeds (500 mm/min to 4500 mm/min) and different axes. The speed range chosen is specific to the 3D printer. The G-code for training phase consists of movement in just one axis (X, Y, and Z), two axes (XY, XZ, and YZ), and all three axes (XYZ). The audio signal corresponding to each of these G-codes is pre-processed and labeled for training the learning algorithms. The total length of audio recorded for training is 1 hour 48 minutes. The total numbers of features extracted is 109 with the window size of 50 ms. For spectral subtraction using STFT, we used the frequency range of 70 Hz to 10 kHz to extract the features. We found that this range is sufficient for the given printer. For regression model, we have used *Decision Trees*, *boosted using Gradient Boosting algorithm*, whereas for the classification model we have used *Decision Tree Classifier*, *boosted using AdaBoost algorithm*. We have trained the learning algorithms and have performed K-fold cross validation, with k = 3, to test the efficiency of the learners as well as to avoid over or under fitting of the learning algorithms. In our experiment, regression model is trained only for the nozzle movements in the X and Y directions. This is because the Z motor moves one layer at a time, and the amount it moves is always fixed (the layer height) for a given object.

5.4.1 Classification Models:

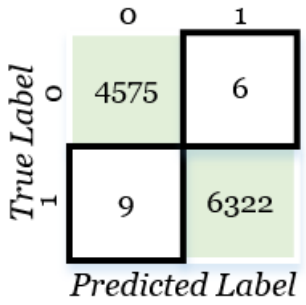
Table 1 shows the accuracy of the various classifiers. We can see that the classification accuracy is higher for simple single axis movement classification compared to complex two

axis movements. The accuracy of the classifiers are calculated by observing the confusion matrix which is shown in Figure 11.

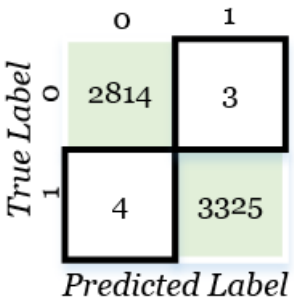
Table 1: Accuracy of the Classification Models

Classifier	Classifying	Accuracy
Φ1	Z|Z'	99.86%
Φ2	1D|2D	99.88%
Φ3	X|Y	99.93%
Φ4	XY_{same}|XY_{Diff}	98.89 %

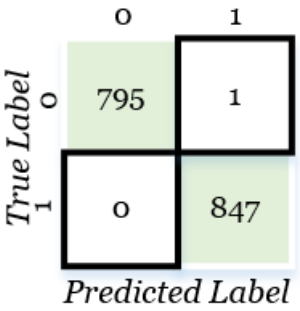
Using Equation (15), we take the true negative and true positive rates and calculate the accuracy of the classifiers.



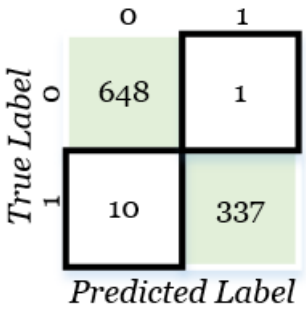
i) Confusion Matrix for Z|Z'



ii) Confusion Matrix for 1D|2D



iii) Confusion Matrix for X|Y



iv) Confusion Matrix for XY Same and Different Speed

Figure 11: Confusion Matrix for Different Classifiers

For measuring the accuracy of the classifiers, Receiver Operating Characteristics (ROC) curves are also analyzed. The classifiers capability to separate the two classes is high if the graph lies closer to the upper right corner. This region corresponds to 100% sensitivity (zero false negatives) and 100% specificity (zero false positives). As the collection of classifiers are arranged in a hierarchy, the bottleneck in terms of accuracy is the longest path followed while making the decision. In this case, the longest path involves all the classifiers.

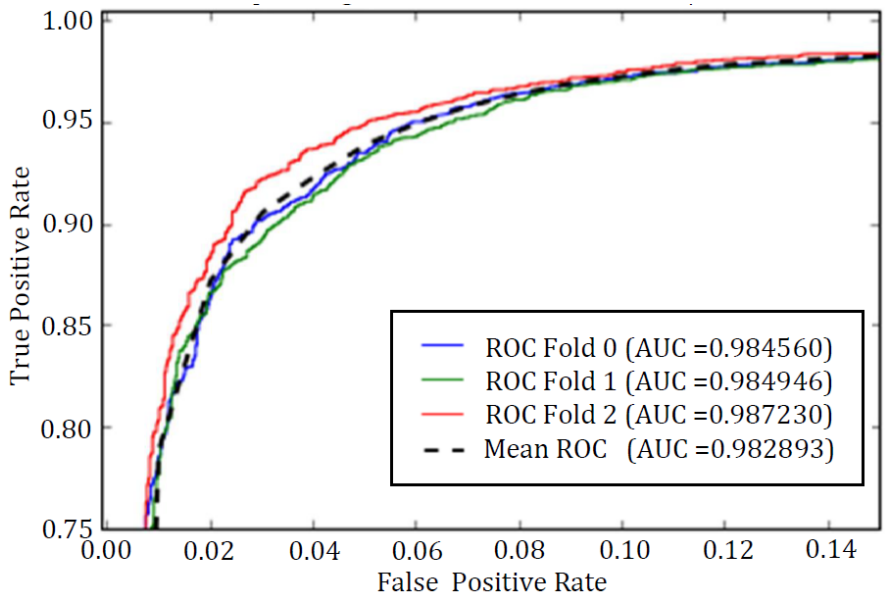


Figure 12: Receiver Operating Characteristic Curve for 1D|2D Classifier

In Figure 12, ROC curve for classifying the movement in either one dimensional (just X or Y axis) or two dimensional (both X and Y together) axis is presented. This classification will allow us to determine if we need to use single axis classifier (X or Y) or the classifier that determines movement of nozzle in both XY axis together with either same speed or different speed. We can see that the curve has quite high Area under the Curve (AUC).

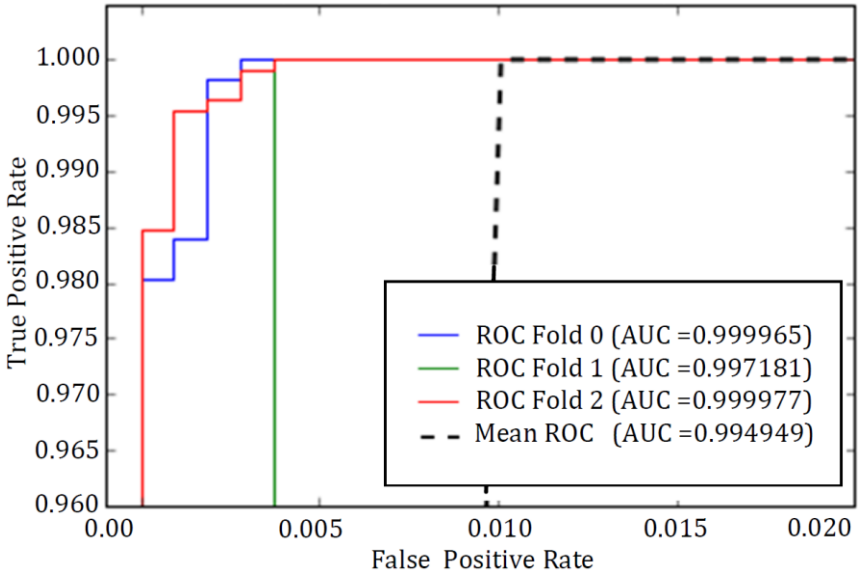


Figure 13: Receiver Operating Characteristic Curve for X|Y Classifier

In Figure 13, the ROC curve for the classifier that separates the movement in either X or Y axis is presented. Compared to two axis separation like Figure 12 and Figure 14 (see below),

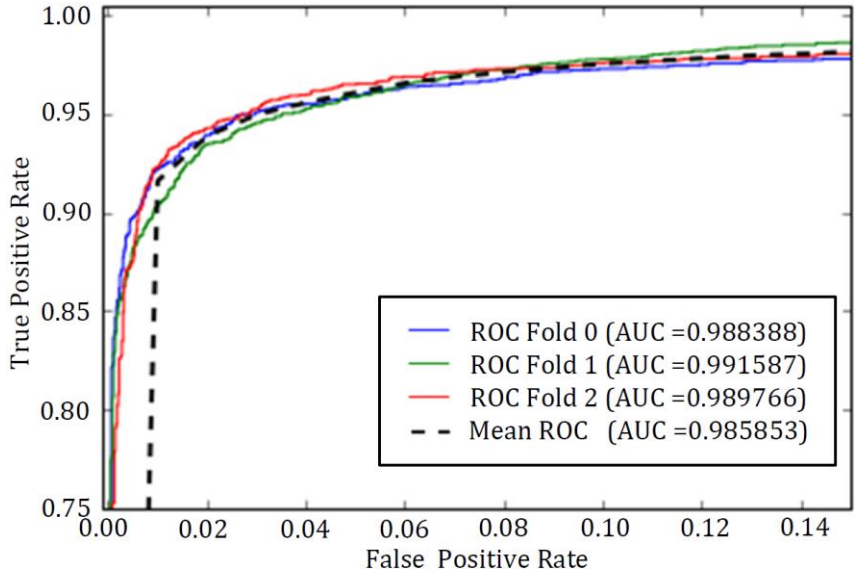


Figure 14: Receiver Operating Characteristic Curve for $XY_{same}|XY_{diff}$ Classifier

the AUC for X or Y movement classification is much higher.

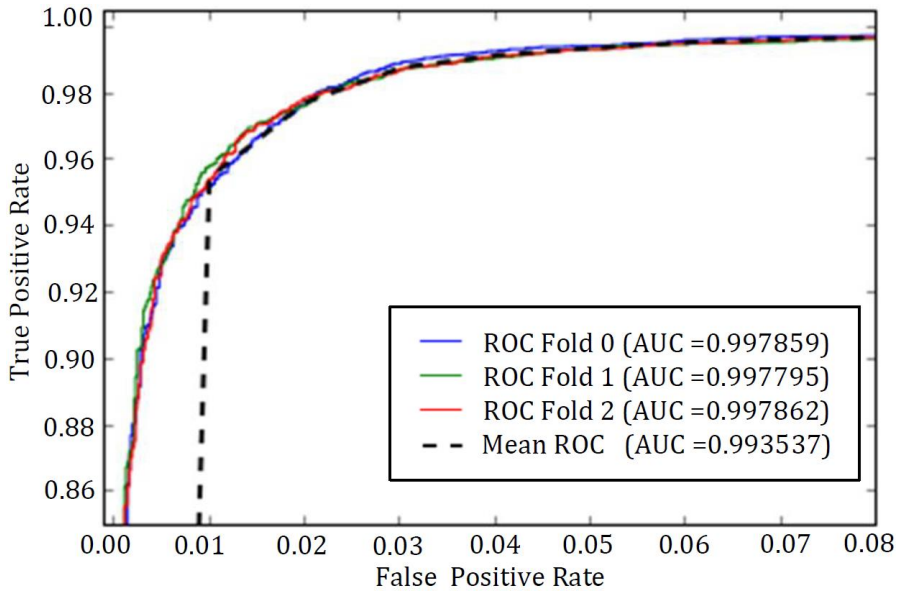


Figure 15: Receiver Operating Characteristic Curve for Z|Z' Classifier

From Figures 12-15, it can be observed that the classifiers have a high sensitivity and specificity with a high Area under the Curve. It means the different classes can be accurately classified based on the observed leakage from the channel $L \rightarrow O$. The AUC for the classifier classifying whether the movement in X and Y axis together with either same speed or different speed is comparatively less than other AUCs. This is intuitive as, in multiple axis movement, separation of individual movement is difficult.

5.4.2 Regression Models

Table 2: Accuracy of the Regression Models

Regression Model	Movement Axis	MSE (Normalized)	Mean Absolute Error (mm/minute)
X	Only X	0.00616	10.1217
Y	Only Y	0.01874	25.5094
X	X and Y	0.16580	150.3374
Y	X and Y	0.42900	314.2519

The accuracy of the regression model is measured in terms of Mean Square Error (MSE) with the data normalized with zero mean and unit variance. We have also presented the mean absolute error to understand how the speed prediction varies from the real speed. From Table 2, we can see that the MSE is relatively higher for the value predicted by the regression model for the motion in Y axis when the motion is occurring at two axes. However, this error can be removed during the post-processing stage as the travel feed rate is generally similar between consecutive frames in each layer of printing. Hence, we can determine the speed of the Y motor if we know the speed of the X motor. Moreover, while printing, most of the time, the range of printing travel feed-rate varies in a narrower range. Hence, we can improve the accuracy of the regression model, if we train it around narrower ranges. This can be done by using unsupervised learning to separate the group of ranges and then using supervised learning algorithms to perform speed prediction in finer resolution.

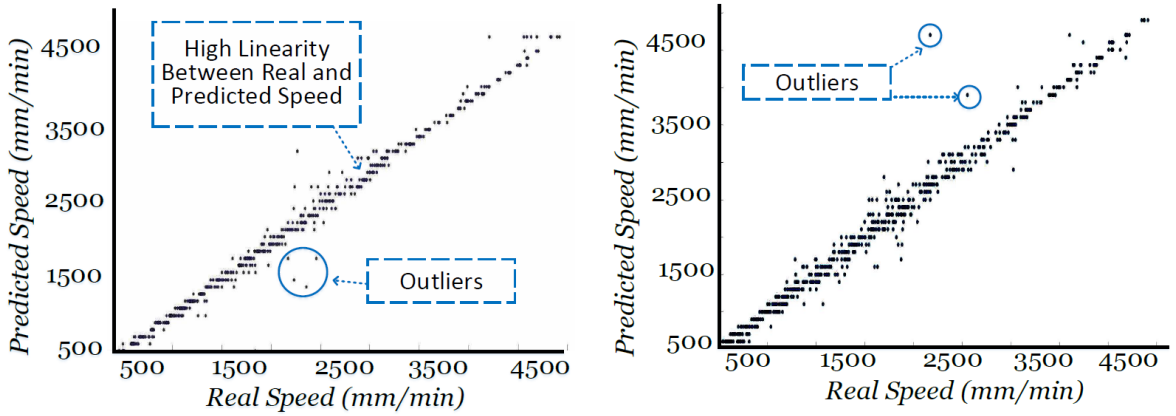


Figure 16: Prediction Results for regression Models in Single Axis

In Figure 16, we can see that there is a linear relationship between the real speed and the predicted speed computed by the regression model.

5.4.3 Direction Prediction Model

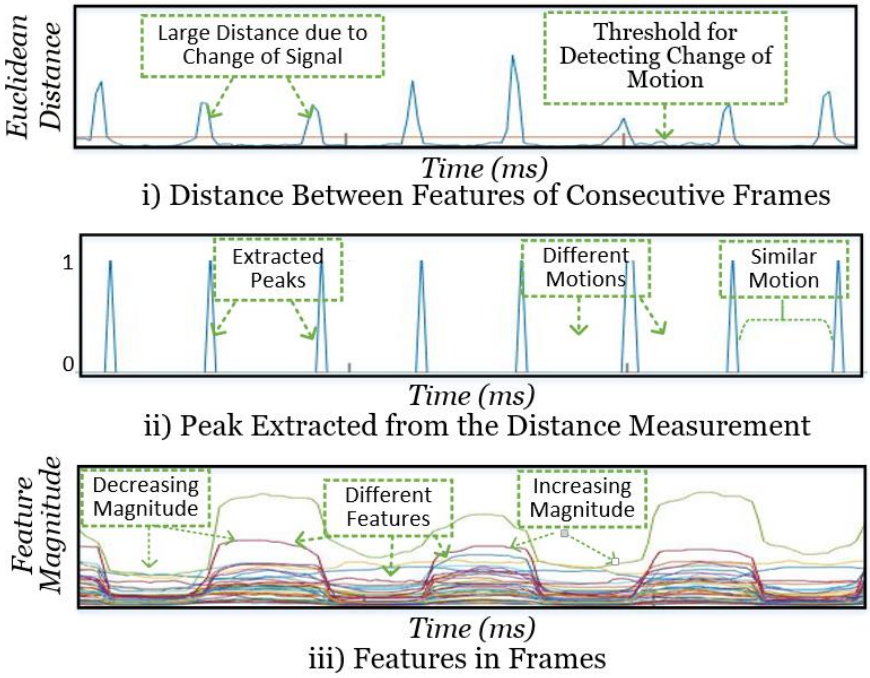


Figure 17: Feature Segmentation and Direction Prediction

Figure 17 shows the feature comparison conducted for the audio recorded while the 3D printer is printing an object. We can observe when the nozzle changes its direction by analyzing the distance of features between successive frames. Peaks are extracted by applying the threshold obtained during the training phase. A value higher than the threshold is 1, otherwise 0. Also in Figure 17 (iii), we can see that the magnitude of the features vary according to direction of movement of the nozzle. Direction prediction model uses this information to determine the direction in each axis for the given line segment (positive or negative direction in X or Y axis).

CHAPTER 6: RESULTS FOR TEST OBJECTS

In order to test our attack model, we define various benchmark parameters which affect the accuracy of the attack model as follows:

Speed of Printing: The fixed frame rate affects the temporal and spectral features extracted from the audio. With the increase in the speed, faster rate of change of spectral features will not be captured and this can degrade the performance of the attack model. Hence, speed of printing is varied to test the accuracy of the attack model.

Dimension of the Object: With smaller objects, shorter nozzle movements are present. To represent these shorter movements, temporal resolution of the features have to be increased by making the frame size smaller. To test our attack model with smaller objects, we have varied the size of the object being printed.

Complexity of the Object: Complex object incorporates movement in more than one axis. Hence, to increase the complexity of the object being created, we have tested the acoustic model with shapes consisting of simultaneous multiple axis movement, such as triangle.

In order to provide the result in a meaningful manner, instead of calculating the mean error square error, in this section, we have calculated the Mean Absolute Percentage Error (MAPE) for the distance prediction.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \left| \frac{A_t - P_t}{A_t} \right| \quad (30)$$

Where $\boldsymbol{A_t}$ is the actual speed and $\boldsymbol{P_t}$ is the speed predicted by the attack model. Since the frame size (50 ms) is same for all the features, distance calculation error will also be given

by equation 30. The classification accuracy is calculated as the total correct prediction made out of total data passed to the classifier.

Table 3: Test Results for Square and Triangle

	Dimension (mm)	Speed (mm/min)	Regression MAPE (%)	Classification Accuracy (%)	Classification Accuracy App I (%)	Classification Accuracy App II (%)	Regression MAPE App II (%)
Square (side)	20	900	5.69	88.57	98.55	98.55	3.13
		1200	9.53	87.35	96.18	97.13	5.40
		1500	14.76	73.76	87.58	96.18	10.94
		1700	26.17	53.66	67.88	67.88	25.24
	10	900	5.82	80.42	97.20	97.20	3.46
		1200	6.63	77.27	95.51	95.51	3.59
		1500	17.53	63.59	85.32	88.12	15.50
		1700	29.78	51.22	65.22	65.22	24.44
	5	900	13.72	55.71	75.11	75.11	10.16
		1200	27.02	53.97	56.97	56.97	26.388
		1500	30.904	51.43	54.22	54.22	29.30
		1700	39.80	47.78	46.67	46.67	31.88
Triangle (base, height)	30,20	900	6.73	85.72	96.77	97.79	3.29
		1200	8.70	81.20	96.69	96.69	4.70
		1500	11.06	71.58	88.68	92.87	10.17
		1700	18.73	61.20	85.44	85.44	15.84
	20,20	900	6.82	84.49	90.74	95.56	3.44
		1200	9.69	75.34	89.31	93.88	6.12
		1500	18.34	73.47	76.67	76.67	12.52
		1700	32.30	52.33	70.77	70.77	28.96
	10,5	900	28.03	60.33	62.33	65.24	23.28
		1200	32.45	55.72	59.86	59.86	30.66
		1500	45.89	54.44	55.54	55.54	40.54
		1700	55.74	50.55	51.43	51.43	58.74
Average			**20.91**	**66.29**	**76.04**	**78.35**	**17.82**

App I: After Post Processing Stage I *App II: After Post Processing Stage II*

For the different parameters used to test the accuracy of the attack model, Table 3 consists of the corresponding results. We can see that the average classification accuracy and regression MAPE before the post-processing stage are 66.29% and 20.91% respectively. Whereas, after post-processing stages, the classification accuracy is 78.35%, and the regression MAPE is 17.82%. We can observe that the post-processing stages have improved the accuracy for the object reconstruction.

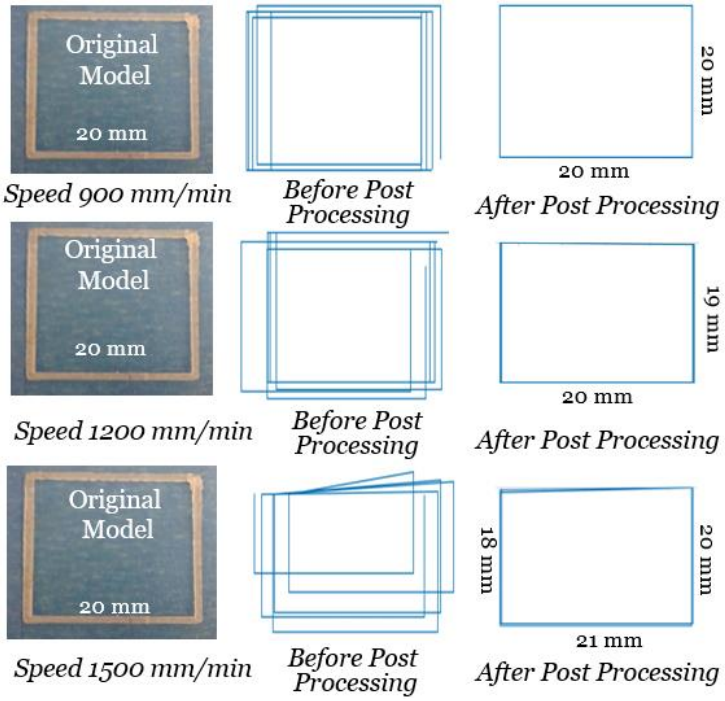


Figure 18: Reconstruction of Square

6.1 Reconstruction of a Square

A square incorporates movements of stepper motors in all axis, however, one at a time. From Table 3, we can see that the accuracy of the classifier for reconstructing the G-code is as high as 88.57% with MAPE of just 5.69%. After post-processing stages the same accuracy has

been increased to 98.55% for the classifier with MAPE of just 3.13%. We can also observe that as the travel feed-rate increases to 1700 mm/min, the accuracy of the classifier and the regression model decreases. Also for short movements such as 5mm, the accuracy of the attack model decreases. Figure 18 shows the square reconstructed by the attack model for a side length of 20 mm.

6.2 Reconstruction of a Triangle

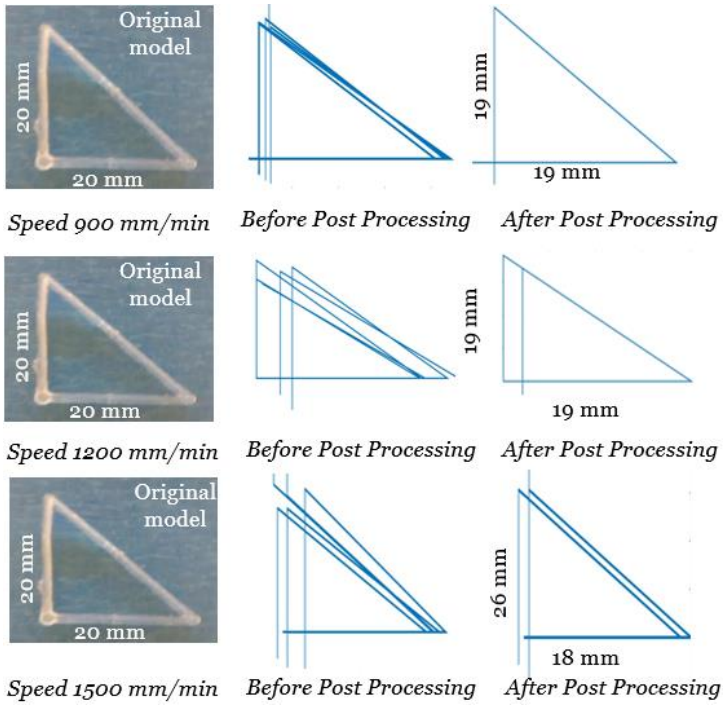


Figure 19: Reconstruction of Triangle

While constructing a triangle, both the X and Y stepper motors have to move at the same time, this affects the reconstruction accuracy of the attack model. From Table 3, we can see that the accuracy of the classifier of the attack model is as high as 85.72% with MAPE of just 6.73% before post-processing and after post-processing they are 97.79% and 3.29%

respectively. As expected, the accuracy of the learning algorithms decreases with increasing speed and decreasing length of the movement. Also, classification and regression accuracy of reconstructing triangle is less compared to square. The G-code reconstructed for one layer of the triangle, with travel feed-rate of 900 mm/min and base and height of 20mm each, is presented in Figure 20. From Figure 19, we can see that the reconstructed shape of the triangle still has some misalignment and wrong direction prediction even after post-processing.

```
;Start G-code
;Initialize 3D Printer
;First Layer
G1 F900  X20 Y0  Z0.4 E1
G1 F900  X0  Y20 Z0.4 E2
G1 F900  X0  Y0  Z0.4 E3
G1 F4500 X0  Y0  Z0.8 E3
;Second Layer
.
;end G-code
```

i) Test G-code for Triangle

```
;Start G-code
;Initialize 3D Printer
;First Layer
G1 F892.5   X20.194 Y0     Z0.4 E1
G1 F899.146 X1.194  Y19.03 Z0.4 E2.2
G1 F905.75  X1.194  Y0.04  Z0.4 E3.2
G1 F4510.4  X0      Y0     Z0.8 E3.2
;Second Layer
.
;end G-code
```

ii) Reconstructed G-code

Figure 20: Partial Reconstructed G-code for Triangle

6.3 Complex Test Object

As a test case for combination of shapes, we have printed an object representing the outline of a key at 900 mm/min travel feed-rate. The classification accuracy obtained before the post-processing is 83.21% and the regression MAPE is 9.15%. After the post-processing, the classification accuracy obtained is 92.54% and the regression MAPE obtained is 6.35%. The object reconstructed by the attack model is shown in Figure 21. As we can see, before post-processing stage II, there are some miscalculated direction and non-uniform lengths in each of the layers of the object. However, after post-processing stage II, these errors are corrected.

In terms of dimension, we can observe that the reconstructed key varies in length and width compared to the original object. Nevertheless, the general outline of the key is reconstructed accurately. Moreover, the accuracy in terms of the length obtained after the post-processing stage is 89.72%, which is calculated by dividing the difference between the original length and the predicted length of each segment in each of the layers by the total length of all the segments in all the layers.

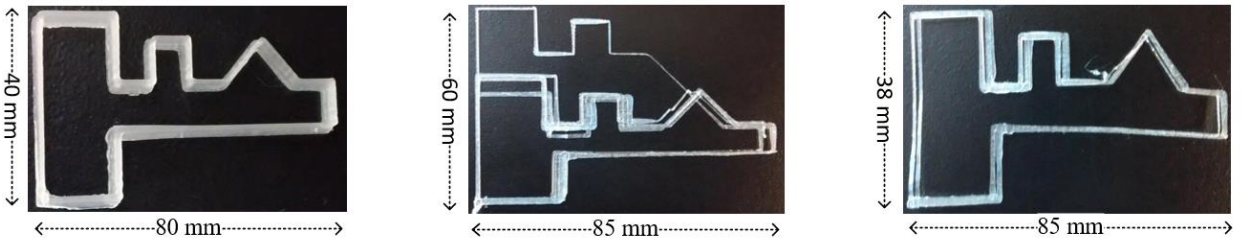


Figure 21: Reconstruction of a Key as a Case Study

The test case, a key, consist of five layers. As shown in figure 21, our attack model is capable of distinguishing the change in the layer, change in the line segment, determine the length of each line segment, and determine the direction of the nozzle movement in each axis.

CHAPTER 7: DISCUSSION

7.1 Limitations of the Attack Methodology

Although we have shown that the attack model is capable of reconstructing the G-code with high accuracy there are some caveat to this methodology described below:

7.1.1 Distance of Microphone

In our current experimental setup, microphone should be placed close enough to be able to detect the variation of the sound received for determining the direction of the motion. The direction prediction model in our attack model relies in the sound/vibration conducted by the motor to the moving part of the printer. If enough sound/vibration is not conducted then the audio device will not be able to capture the variation. In this case, some other sensors can be incorporated along with the audio device, such as proximity sensors, to detect the direction. We leave this for our future work.

7.1.2 Short and Rapid Movements

Due to the fixed frame size incorporated for feature extraction in our experiment, the accuracy of the attack model is reduced for higher speed and smaller dimension. In order to capture smaller movements, the temporal resolution of the features extracted have to be increased by making the frame size smaller. However, for faster speeds, we need larger frame size to increase the frequency resolution for better spectral features. This trade-off

dictates that we should incorporate adaptive frame size to increase the accuracy of the attack model. We will incorporate this in our future work.

7.1.3 Multiple Axis Movement and Noise

The separation of sound source from combination of sound is a well-known problem in speech processing. In our attack model, we have incorporated spectral subtraction to acquire features that are unique to each of the stepper motors. However, there are other methods in separating the sources of sound [53][54]. Incorporating them in the attack model may increase the accuracy of the G-code being reconstructed. We leave this for our future work.

7.2 3D Printer Variation

Table 4: 3D Printers Available in the Market

3D Printer	Motors (Number)	X	Y	Z	E
Makergear M2 (www.makergear.com)	4	1	1	1	1
FlashForge (www.flashforge-usa.com)	5	1	1	1	2
Ultimaker 2 (www.ultimaker.com)	4	1	1	1	1
LulzBot TAZ (www.lulzbot.com)	4	1	1	1	1

As we can see in Table 4, most of the Fused Deposition Modeling (FDM) based 3D printers available in the market consists of an equal number of stepper motors. These models also have Z motor to control the layer height. They only vary in speed and the resolution of printing. Our attack methodology can be used on any of these printers. However, this attack

methodology needs to be tested against other types of printers based on selective laser sintering and selective laser melting, as their structure varies from the FDM based 3D printers. We leave this for our future work.

7.3 Multiple Side-Channel Analysis

In this thesis work, we analyzed the acoustic side-channel, however, as mentioned earlier, cyber-physical additive manufacturing system have various side-channels through which information leakage is possible. It is possible that combination of these side-channels can cumulatively leak more information through the analog emissions. Hence, it is imperative to design more complex attack models that consider multiple side-channels for multivariate analysis of information leakage. However, this multivariate analysis can follow the same methodology as described in this thesis.

7.4 Counter Measures

7.4.1 Similar Loads on Each Motor

If each of the stepper motor moves equal load, then the acoustic features emanated will be similar for all the motors. In this scenario, the attack model will not be able to distinguish the movement of one motor from another. However, making loads equal in each motor requires restructuring the design of the printers. Hence, printer design methodology has to consider impact of structural design on side-channel leakage as a parameter for designing secure additive manufacturing system.

7.4.2 G-code variation

To make the G-code reconstruction process harder, randomness in the movements of the motors can be incorporated in the G-code. However, this countermeasure will delay the total printing time by adding redundancy in the G-code while improving security against the acoustic side-channel attack. Further research is required for designing efficient algorithms that slice the 3D models to increase the entropy of the signals leaked through the side-channel.

7.4.3 Leakage Aware CAD and CAM Tools

Rather than providing run-time solutions, design time solution can be incorporated by make the computer aided design and computer aided manufacturing tools aware about the leakage from the side-channels. Designing such tools will require a framework that is capable to predict the amount of information leakage from the side-channels given the specific mechanical structure of the 3D printer and the structure of the 3D object.

CHAPTER 8: CONCLUSION

In this thesis, we have presented a novel acoustic side-channel attack model for cyber-physical additive manufacturing system to reconstruct cyber domain data, which includes IP. Additionally, we have tested our attack model with a state-of-the-art 3D printer to reconstruct objects with different benchmark parameters such as speed, dimension, and complexity. We have successfully performed the acoustic side-channel attack with an average axis prediction accuracy of 66.29%, and average length prediction error of 20.91%. Furthermore, with post-processing we have achieved a moderately high average axis prediction accuracy of 78.35% and average length prediction error of 17.82%. Specifically, our attack model has achieved a high axis prediction accuracy of 92.54% and a small length prediction error of 6.35%, when testing it with a complex object such as a key. In addition to testing and validating our attack model, we also have discussed some of its limitations and countermeasures. Nonetheless, our work serves as a proof of concept of a serious physical-to-cyber domain attack, which acquires and utilizes side-channel information (such as acoustic signals) from additive manufacturing systems) to steal the valuable cyber domain data.

REFERENCE

[1] R. Rajkumar, I. L. I. Lee, L. S. L. Sha, and J. Stankovic, "Cyber-physical systems: The next computing revolution," in *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, 2010, pp. 0–5.

[2] K. V. Wong and A. Hernandez, "A Review of Additive Manufacturing," *ISRN Mech. Eng.*, vol. 2012, pp. 1–10, 2012.

[3] F. Ning, W. Cong, J. Qiu, J. Wei, and S. Wang, "Additive manufacturing of carbon fiber reinforced thermoplastic composites using fused deposition modeling," *Compos. Part B Eng.*, vol. 80, pp. 369–378, 2015.

[4] B. Leukers *et al.*, "Hydroxyapatite scaffolds for bone tissue engineering made by 3D printing," in *Journal of Materials Science: Materials in Medicine*, 2005, vol. 16, no. 12, pp. 1121–1124.

[5] M. & A. Thryft, Ann R. (senior Technical Editor, "NASA Builds 3D Printer for Space.," *Des. News*, vol. 68, no. 8, pp. p27-28, 2013.

[6] M. Tomlin and J. Meyer, "Topology Optimization of an Additive Layer Manufactured (ALM) Aerospace Part," *7th Altair CAE Technol. Conf. 2011*, pp. 1–9, 2011.

[7] C. A. Giffi, B. Gangula, and P. Illinda, "Additive manufacturing: 3D opportunity for the automotive industry," *Deloitte University Press*, 2014. [Online]. Available: http://dupress.deloitte.com/dup-us-en/focus/3d-opportunity/additive-manufacturing-3d-opportunity-in-automotive.html.

[8] P. A. Kobryn and S. L. Semiatin, "The laser additive manufacture of Ti-6Al-4V," *Jom*, vol. 53, no. 9, pp. 40–42, 2001.

[9] B. Short, "Quality Metal Additive Manufacturing PowerPoint Presentation,PPT - DocSlides," *www.navy.mil*, 2015. [Online]. Available: http://www.docslides.com/sherrill-nordquist/quality-metal-additive-manufacturing.

[10] T. Lewis, "How NASA Is Launching 3D Printing Into Space," *www.space.com*, 2014. [Online]. Available: http://www.space.com/24599-nasa-launches-3d-printing-in-space.html.

[11] T. Wohlers and T. Caffrey, *Wohlers Report 2015: 3D Printing and Additive Manufacturing State of the Industry Annual Worldwide Progress Report*. 2014.

[12] R. Kemp, "Fourth industrial revolution," *Lawyer, ISSN-e 0953-7902, Vol. 31, Nº. 21 (6/6/2016), 2016*, vol. 31, no. 21, p. 12, 2016.

[13] P. Daugherty, P. Banerjee, W. Negm, and A. E. Alter, "Driving Unconventional Growth through the Industrial Internet of Things," 2015.

[14] L. J. Wells, J. A. Camelio, C. B. Williams, and J. White, "Cyber-physical security challenges in manufacturing systems," *Manuf. Lett.*, vol. 2, no. 1, pp. 74–77, 2013.

[15] M. Yampolskiy, L. Schutzle, U. Vaidya, and A. Yasinsac, "SECURITY CHALLENGES OF ADDITIVE MANUFACTURING WITH METALS AND ALLOYS," in *CRITICAL INFRASTRUCTURE PROTECTION IX*, 2015, vol. 466, pp. 169–183.

[16] M. K. Daly, "Industrial Cyber Security in Additive Manufacturing," 2014. [Online].

Available: http://www.additivemanufacturingfordefense.com/industrial-cyber-security-in-additive-manufacturing-mloc-f-ty-m#.
[17] J. Hughes and G. Cybenko, “Three tenets for secure cyber-physical system design and assessment,” *SPIE Def. + Secur.*, p. 90970A, 2014.
[18] T. McDermott, “Manufacturing – A Persistent and Prime Cyber Attack Target | CohnReznick,” *cohn reznick*, 2014. [Online]. Available: https://www.cohnreznick.com/manufacturing-persistent-and-prime-cyber-attack-target.
[19] J. Bisceglie and M. McGrath, “White Paper – Cybersecurity for Advanced Manufacturing,” 2014.
[20] “Intellectual Property and the U.S. Economy: Industries in Focus | Economics & Statistics Administration,” 2012. [Online]. Available: http://www.esa.doc.gov/reports/intellectual-property-and-us-economy-industries-focus.
[21] M. Hvistendahl, “3D printers vulnerable to spying,” *Science (80-.).*, vol. 352, no. 6282, 2016.
[22] M. Yampolskiy, T. R. Andel, J. T. McDonald, W. B. Glisson, and A. Yasinsac, “Intellectual Property Protection in Additive Layer Manufacturing,” in *Proceedings of the 4th Program Protection and Reverse Engineering Workshop on 4th Program Protection and Reverse Engineering Workshop - PPREW-4*, 2014, pp. 1–9.
[23] I. Gibson, D. W. Rosen, and B. Stucker, “Additive Manufacturing Technologies,” *Addit. Manuf. Technol. Rapid Prototyp. to Direct Digit. Manuf.*, pp. 1–459, 2010.
[24] W. Ashford, “21% of manufacturers hit by intellectual property theft,” *Computer Weekly*, 2014. [Online]. Available: http://www.computerweekly.com/news/2240226840/IP-theft-hit-21-of-manufacturers-in-past-year-study-shows. [Accessed: 23-Oct-2016].
[25] K. ZETTER, “Report Details Hacks Targeting Google, Others | WIRED,” *Wired*, 2010. [Online]. Available: https://www.wired.com/2010/02/apt-hacks/. [Accessed: 23-Oct-2016].
[26] I. Warfare Monitor, “Information Warfare Monitor Investigating a Cyber Espionage Network,” 2009.
[27] F. X. Standaert, T. G. Malkin, and M. Yung, “A unified framework for the analysis of side-channel key recovery attacks,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2009, vol. 5479 LNCS, pp. 443–461.
[28] et all Rogério dos Santos Alves; Alex Soares de Souza, *Google Sketchup for 3D Printing*, no. 1. 2014.
[29] M. Muchmore, *Adobe Photoshop CC 2015*. 2015.
[30] J. D. Hiller and H. Lipson, “STL 2.0: A Proposal for a Universal Multi-Material Additive Manufacturing File Format,” *Proc. 20th Solid Free. Fabr. Symp.*, no. 1, pp. 266–278, 2009.
[31] T. R. Holbrook and L. Osborn, “Digital Patent Infringement in an Era of 3D Printing,” *Leg. Stud. Res. Pap. Ser. Res. Pap.*, vol. 48, no. 1319, pp. 1318–2015, 2014.
[32] L. Sturm, C. Williams, J. Camelio, and J. White, “Cyber-physical vunerabilities in

additive manufacturing systems," *Context 7*, 2014.
[33] M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, and C. Sporleder, "Acoustic side-channel attacks on printers," *Proc. 19th USENIX Conf. Secur.*, pp. 20–20, 2010.
[34] E. Toreini, B. Randell, F. Hao, E. Toreini, B. Randell, and F. Hao, "An Acoustic Side Channel Attack on Enigma An Acoustic Side Channel Attack on Enigma," 2015.
[35] A. Davis, M. Rubinstein, N. Wadhwa, G. J. Mysore, F. Durand, and W. T. Freeman, "The Visual Microphone : Passive Recovery of Sound from Video," *Siggraph 2014*, pp. 1–10, 2014.
[36] H. Vincent, L. Wells, P. Tarazaga, and J. Camelio, "Trojan Detection and Side-channel Analyses for Cyber-security in Cyber-physical Manufacturing Systems," *Procedia Manuf.*, vol. 1, pp. 77–85, 2015.
[37] D. . Pham and R. . Gault, "A comparison of rapid prototyping technologies," *Int. J. Mach. Tools Manuf.*, vol. 38, no. 10–11, pp. 1257–1287, 1998.
[38] R. Condit, "Stepping Motors Fundamentals," *Microchip Technol.*, vol. AN907, pp. 1–22, 2004.
[39] T. Kenjō and A. Sugawara, *Stepping motors and their microprocessor controls*. Clarendon Press, 1994.
[40] A. Hughes and P. J. Lawrenson, "Electromagnetic damping in stepping motors," *Proc. Inst. Electr. Eng.*, vol. 122, no. 8, p. 819, 1975.
[41] P. L. Timar, *Noise and Vibration of Electrical Machines*. Elsevier Science, 1989.
[42] B. Heller and V. Hamata, *Harmonic field effects in induction machines*. Elsevier Scientific Pub. Co., 1977.
[43] S. J. Yang, *Low-noise electrical motors*. Clarendon Press, 1981.
[44] J. F. Gieras, C. Wang, and J. C. Lai, *Noise of polyphase electric motors*. 2006.
[45] E. C. T. So, R. G. D. Williams, and S. J. Yang, "A simple model to calculate the stator radial vibration of a hybrid stepping motor," in *Conference Record of the 1993 IEEE Industry Applications Conference Twenty-Eighth IAS Annual Meeting*, pp. 122–129.
[46] A. Yadav, "Nyquist-Shannon Sampling Theorem," in *Digital Communication*, 2009, pp. 9–16.
[47] S. Theodoridis and K. Koutroumbas, *An Introduction to Pattern Recognition: A MATLAB Approach, Second Edition*, vol. 11. 2006.
[48] B. Logan, "Mel Frequency Cepstral Coefficients for Music Modeling," *Int. Symp. Music Inf. Retr.*, vol. 28, p. 11p., 2000.
[49] F. Pedregosa and G. Varoquaux, *Scikit-learn: Machine learning in Python*, vol. 12. 2011.
[50] "Printrbot Simple Kit," 2016. [Online]. Available: https://printrbot.com/shop/simple-metal-kit/.
[51] "Zoom H6 Handy Recorder," 2016. [Online]. Available: https://www.zoom-na.com/products/field-video-recording/field-recording/h6-handy-recorder.
[52] T. M. Inc., "MATLAB (R2015b)," *MathWorks Inc.*, 2015.
[53] D. Barry, B. Lawlor, and E. Coyle, "Sound Source Separation: Azimuth Discrimination and Resynthesis," *In Pract.*, pp. 5–10, 2004.
[54] M. S. Pedersen, J. Larsen, U. Kjems, and L. C. Parra, "A Survey of Convolutive Blind Source Separation Methods," *Speech Commun.*, pp. 1–34, 2007.