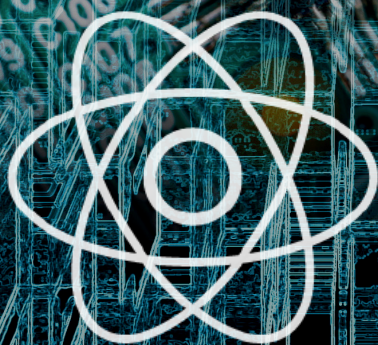


JOURNAL OF ENGINEERING RESEARCH & SCIENCES

JENRS



www.jenrs.com
ISSN: 2831-4085

Volume 3 Issue 7
July 2024

EDITORIAL BOARD

Editor-in-Chief

Prof. Paul Andrew
Universidade De São Paulo, Brazil

Editorial Board Members

Dr. Jianhang Shi
Department of Chemical and Biomolecular
Engineering, The Ohio State University, USA

Dr. Sonal Agrawal
Rush Alzheimer's Disease Center, Rush
University Medical Center, USA

Dr. Unnati Sunilkumar Shah
Department of Computer Science, Utica
University, USA

Prof. Anle Mu
School of Mechanical and Precision Instrument
Engineering, Xi'an University of Technology,
China

Dr. Qiong Chen
Navigation College, Jimei University, China

Dr. Diego Cristallini
Department of Signal Processing & Imaging
Radar, Fraunhofer FHR, Germany

Dr. Jianhui Li
Molecular Biophysics and Biochemistry, Yale
University, USA

Dr. Lixin Wang
Department of Computer Science, Columbus
State University, USA

Dr. Prabhash Dadhich
Biomedical Research, CellfBio, USA

Dr. Żywiołek Justyna
Faculty of Management, Czestochowa
University of Technology, Poland

Dr. Anna Formica
National Research Council, Istituto di
Analisi dei Sistemi ed Informatica, Italy

Prof. Kamran Iqbal
Department of Systems Engineering,
University of Arkansas Little Rock, USA

Dr. Ramcharan Singh Angom
Biochemistry and Molecular Biology,
Mayo Clinic, USA

Dr. Qichun Zhang
Department of Computer Science,
University of Bradford, UK

Dr. Mingsen Pan
University of Texas at Arlington, USA

Ms. Madhuri Inupakutika
Department of Biological Science,
University of North Texas, USA

Editorial

The current issue of the journal brings together a collection of five research papers that span a range of groundbreaking topics, each addressing critical aspects of modern science and technology. These studies, though varied in their focus, share a common theme of advancing our understanding and capabilities in their respective fields, offering insights that could shape future practices and policies.

The first paper explores the ongoing debate between electric vehicles (EVs) and internal combustion engine (ICE) vehicles, shedding light on the intertwined interests of the automotive and oil industries. As the world grapples with the consequences of climate change, the transition to EVs is positioned as not just an option but a necessity for sustainable human civilization. The authors make a compelling case for the environmental benefits of electric vehicles, including reduced pollution in densely populated urban areas and the potential for EV batteries to serve as a backup for power grids. This paper challenges readers to consider the broader implications of vehicle technology choices and the vested interests that may hinder the shift towards cleaner alternatives [1].

In the realm of cybersecurity, the second paper delves into the complexities of protecting digital environments from cyber threats. With the rise of cyberattacks, safeguarding information systems has never been more crucial. The paper highlights the transformative role of artificial intelligence, specifically machine learning and deep learning, in fortifying cybersecurity measures. By analyzing patterns and predicting potential threats, these advanced technologies are proving invaluable in preempting cyber intrusions. This research underscores the importance of continuous innovation in cybersecurity practices to protect sensitive data and maintain trust in digital infrastructures [2].

The third study presents an in-depth analysis of software-defined networking (SDN), an innovative approach to network architecture that enhances control and operational efficiency. Using the Mininet simulator, the authors evaluate the performance of various network topologies and protocols, offering insights into the optimal configurations for different scenarios. The findings from this research are crucial for network administrators and developers aiming to implement SDN frameworks that maximize performance and reliability. By providing a detailed benchmark of performance metrics, this study contributes significantly to both academic research and practical applications in network management [3].

Moving to the field of medical imaging and diagnosis, the fourth paper introduces a novel approach to detecting keratoconus, a non-inflammatory eye disorder. The authors employ advanced techniques such as principal component analysis (PCA) and an improved recurrent neural network (RNN) with Grey Wolf optimization to enhance the accuracy of early diagnosis. Early detection of keratoconus is vital to prevent complications, especially in patients undergoing refractive surgery. This research not only improves diagnostic capabilities but also opens new avenues for developing more effective treatment plans, thereby enhancing patient outcomes [4].

The fifth paper addresses the challenge of text detection and recognition in multilingual contexts, specifically focusing on the Bengali language. While significant progress has been made in text recognition for languages like English, the unique characteristics of Bengali script pose additional challenges. The authors propose a combination of advanced machine learning models, including Mask-R-CNN, CRNN, and a novel Fast Iterative Nearest Neighbour (Fast INN) algorithm, to achieve high accuracy in text recognition. This research represents a significant step forward in making text recognition technology more inclusive and effective across different languages and scripts [5].

Each of these papers contributes to its field by addressing pressing issues with innovative approaches and thorough research methodologies. Together, they illustrate the diverse challenges and opportunities that modern science and technology present. As we navigate these complexities, it is clear that interdisciplinary research and collaboration are key to driving progress and fostering a more sustainable and technologically advanced future.

References:

- [1] P. Karunakaran, M. Shahril Osman, "Reviewing the Value of Electric Vehicles in Achieving Sustainability," *Journal of Engineering Research and Sciences*, vol. 3, no. 7, pp. 1–10, 2024, doi:10.55708/js0307001.
- [2] R. Khalid, M. Naqi Raza, "A Thorough Examination of the Importance of Machine Learning and Deep Learning Methodologies in the Realm of Cybersecurity: An Exhaustive Analysis," *Journal of Engineering Research and Sciences*, vol. 3, no. 7, pp. 11–22, 2024, doi:10.55708/js0307002.
- [3] N. V. Oikonomou, D. V. Oikonomou, E. Stergiou, D. Liarakis, "Comprehensive Analysis of Software-Defined Networking: Evaluating Performance Across Diverse Topologies and Investigating Topology Discovery Protocols," *Journal of Engineering Research and Sciences*, vol. 3, no. 7, pp. 23–43, 2024, doi:10.55708/js0307003.
- [4] S. Hassan Musa, Q.J. Mohammed Alhaidar, M. Mahdi Borhan Elmi, "Keratoconus Disease Prediction by Utilizing Feature-Based Recurrent Neural Network," *Journal of Engineering Research and Sciences*, vol. 3, no. 7, pp. 44–52, 2024, doi:10.55708/js0307004.
- [5] M. Dutta, D. Tripura, J. Krishna Das, "A Computational Approach for Recognizing Text in Digital and Natural Frames," *Journal of Engineering Research and Sciences*, vol. 3, no. 7, pp. 53–58, 2024, doi:10.55708/js0307005.

Editor-in-chief

Prof. Paul Andrew

CONTENTS

<i>Reviewing the Value of Electric Vehicles in Achieving Sustainability</i> Prashobh Karunakaran, Mohammad Shahril Osman	01
<i>A Thorough Examination of the Importance of Machine Learning and Deep Learning Methodologies in the Realm of Cybersecurity: An Exhaustive Analysis</i> Ramsha Khalid , Muhammad Naqi Raza	11
<i>Comprehensive Analysis of Software-Defined Networking: Evaluating Performance Across Diverse Topologies and Investigating Topology Discovery Protocols</i> Nikolaos V. Oikonomou, Dimitrios V. Oikonomou, Eleftherios Stergiou, Dimitrios Liarokapis	23
<i>Keratoconus Disease Prediction by Utilizing Feature-Based Recurrent Neural Network</i> Saja Hassan Musa, Qaderiya Jaafar Mohammed Alhaidar, Mohammad Mahdi Borhan Elmi	44
<i>A Flourished Approach for Recognizing Text in Digital and Natural Frames</i> Mithun Dutta, Dhonita Tripura, Jugal Krishna Das	53

Reviewing the Value of Electric Vehicles in Achieving Sustainability

Prashobh Karunakaran*, Mohammad Shahril Osman

University of Technology Sarawak (UTS), SET CRISD, Sibul, 96000, Malaysia

*Corresponding Author: Prashobh Karunakaran, 19 Greenwood Park Phase 4, 9th Mile Penrissen Road, Kuching, Malaysia, 0128879578, prashobh.karunakaran@gmail.com

ABSTRACT: This paper aims to narrow the gap of the narratives blasted out in the media (including social media) about electric cars versus the conventional way humans have been transported over the last 100 years. The ICE industry is closely connected to the O & G because 64 % of the output of the O & G industry is utilized for the transportation industry, which ranges from motorcycles, cars, trucks, ships to airplanes. Therefore, these two large industries have a motive to curtail the expansion of the electric vehicles industry. This paper explains why the change from ICE to BEV is needed for the sustainability of human civilization because climate change has been proven to be linked to human activities. Without tailpipe emissions, electric vehicles can immediately clean up pollution in the largest cities of earth where 56 % of humanity lives. The batteries of electric vehicles can also become a citizen sponsored backup battery for the electric power grids, thereby saving even more pollution from the environment. Such a backup for the power grid is necessary for its ability to smoothen the sudden and unexpected spikes in electric consumption or tripping of generators in the power grid.

KEYWORDS: Battery Electric Vehicles (BEV), Fuel Cell Electric Vehicles (FCEV), Internal Combustion Engines (ICE), sustainability, pollution

1. Introduction

Humans walked or ran to their destinations for the longest time. Then they learnt to domesticate horses and cattle to help in their transportation. Then 171 years ago, in 1853 Eugenio Barsanti and Felece Matteucci invented the ICE (internal combustion engine) and like a slow conflagration, ICE went all round the earth to help humanity in actuation and transportation [1].

The internal combustion engine took the idea from the gun or cannon which is to push something out of a barrel, but it was a linear motion. So, weights in the crankshaft were used to smoothen the jerks at the end of linear motions. This was developed over 100 years to almost perfection [2].

Comparatively, electric motors actuate with circular motion, therefore not needing weights to overcome jerks [3]. The electric vehicle (EV) is driven with an electric motor, which was first demonstrated by Micheal Faraday in 1821. As ICEs were being popularized by the likes of

Henry Ford and JD Rockefeller, there were people also pushing for transportation to be driven by electric motors. Actually, Edison wanted to use biofuels to help farmers but gave up and started to use Rockefeller's fuel. JD Rockefeller was a clever businessman who built petrol stations at regular intervals such that people won't run out of fuel and the market went for it and EVs lost their market share [4]. The much faster refilling of fuel into ICE compared to EV also favored ICE; today this problem is gradually being solved with transformers increasing the voltage at which the BEV (battery electric vehicle) can be charged [5].

The Japanese then took precision to the next level mainly as an outcome of Dr. Deming teaching them statistics after WWII. Dr. Deming was a PhD in Physics who helped develop statistics under Dr. Walter Andrew Shewhart of the Bell Laboratories in New York. They came up with the use of statistics in the manufacture of military equipment in the USA. Guns used to get jammed in battlefields and introducing statistics into armament

factories reduced incidences of these happening greatly. Dr. Deming was later invited to Japan by General MacArthur to help with the first post war census. Then in 1950 the general also invited Dr. Deming to the Japanese Union of Scientists and Engineers (JUSE) to talk about statistical process control which is the first time Japanese manufacturers got interested in the use of statistics in manufacturing. Dr. Deming also taught the Japanese manufacturers that as a company improves its quality, expenses will reduce, and market share will increase [6]. When electric cars first came out of Tesla, Inc., the Japanese were working on hybrids which had two engines: an ICE and an electric motor. This made them heavy.

The story of electric cars is just like that of the bow and arrow industry of old. There were a whole lot of institutions and respected teachers (shifus) in the making of the bows, the strings, the assembling, the shifus to teach the art of shooting (Kung Fu, kalaripayattu etc.) Then someone came up with the gun. People in the long chain of institutions of the bow and arrow industry have vested interest in keeping guns at bay. But when countries that adopted guns started conquering the South despite them knowing how to build it (they had firecrackers), all started adopting guns and cannons. Hybrids are like crossbows.

Today in 2024, there is a heated battle between Internal Combustion Engine (ICE), Battery Electric Vehicle (BEV) and Fuel Cell Electric Vehicle (FCEV) which are running on hydrogen. The ICE industry is linked to the Oil and Gas (O & G) industry [7] because 64% of the output of the O & G industry is used for transportation. If humanity switched to BEV, two huge industries would collapse, the O & G and ICE industries. Today the O & G is still one of the most lucrative industries on earth, making up 3.8% of the world's economy [8]. In the USA, the bureau of statistics revealed that 118 thousand people work for the O & G industry and generated a revenue of \$333 billion, which is a per capita income of \$2,822,034 [9]. Therefore, the O & G industry has a lot of capital to delay the advent of other modes of transportation that do not use their products. One option is to use hydrogen, 95% of which is derived from fossil fuels, thereby keeping the O & G industry alive [10]. But CO₂ emission from the SMR (steam methane reforming process) produces 9 kg of CO₂ per kg of H₂. Usually natural gas (70-90% methane) is sent directly into the process chamber and heated with steam. Natural gas got its name in the 1800s to differentiate it from coal gas which is derived from heating coal. Gas that comes out of the ground is termed natural gas. Natural gas, like fossil fuels, is derived from decaying organic matter. The volume it occupies makes it impractical to be

used without compressing to Liquefied Natural Gas (LNG), unless as in Bintulu, Malaysia where the gas from offshore oil rigs is sent directly into gas turbines (GTs). Similarly, the USA gets 40 % of her electricity from GTs, most of which are powered directly from fracking plants, especially in the Permian Basin in Texas or the Marcellus Shale in Pennsylvania [11].

Today the biggest fight against the BEV is the hydrogen cars, mostly of the FCEV type. This is because the O & G industry knows that humanity is poised to reduce pollution, especially the 56 % of humanity who live in large cities, who do not need to see any CO₂ level data. They are constantly breathing and washing the combustion fumes from their bodies and clothes every day [12].

It must be noted that all major industries have long gone electric. The huge trucks in the mines in Australia (where a standing human is only as high as half of the wheel), the largest cranes in the O & G, cranes in ports, trains, the largest prime movers in factories and the prime movers of the largest ships [13].

For example, the Sejinkat Port of Kuching has cranes, two of which were hydraulic and two used electric motors. All four cranes were installed at the same time, but the hydraulic cranes failed very fast while the motor crane has been running since 1987 when the port was built. Motors, especially induction motors designed by Nikola Tesla are robust. A squirrel cage rotor and a stator with copper coils is just too simple to fail. The only source of failure are the bearings and the varnish surrounding the copper coils. But bearing companies, the likes of SKF have perfected it to a high extent over 117 years (since 1907) and copper coil varnish of Class H has been known to enable even large motors to run continually for long spans of time. The induction motor in the Western Digital factory of Kuching has been running the compressor since 1995 (27 years). This induction motor utilizes 600 A per phase, or 1800 A [14].

Prime movers of ships for example must overcome the force of waves of the ocean. ICE is not good at handling sudden changes in the speed caused by the powerful waves. But a simple rotor surrounded by copper coils in the stator can handle it much better.

One of the biggest pros for BEV is that, if each householder has a BEV or many plugged in, they will form a citizen financed backup battery for the electric power grid. When there is a sudden cloud cover over solar farms or a sudden no-wind situation in wind farms, the grid can take power from people's electric cars.

The above statement is for the average human on

earth. But in actuality the power grid will need batteries for more reasons than that. Being a Control Room engineer for the electric power grid of Sarawak, Malaysia, it was noted that engines all over the grid have their accelerator pressed 30 % more than the demand for power. This is called the Spinning Reserve. This is because even if one of the engines in a power station trip, there will be a sudden (millisecond) demand greater than supply situation, causing a sudden greater draw from the stator coils of generators. This will also make them stronger electromagnets which will slow the speed of the rotor which will cause the frequency to drop. The frequency of the voltage wave is the speed of the spinning of the generator stator. Frequency drop is very dangerous. To simplify things, there are three stator coils spaced at 120° apart in the stator of generators (which generate the three phases of power). Actually, there are six coils, each phase coil will make a clockwise turn at one end and an anticlockwise turn at the opposite end and end up in a star point with the other two phases. Therefore, one end is a star point and the other end supplies the load of the country. The rotor is an electromagnet energized with DC. As the rotor's north pole passes the clockwise coil, its south pole will pass the anticlockwise coil, thereby providing double the current generated in a phase line. The generators generally output at 11 kV – 15 kV but this is then stepped up to 275 kV and is then joined to all other generators in the grid which may be 1000 km away. To join generators in a grid, all generators must be synchronized. Meaning if all are ICE powered generators, if the rotor electromagnet's north pole is passing the L1 phase coil in one generator, it must pass L1 in all other generators [15].

If the rotor electromagnet's north pole is passing L1 in one generator and in the second generator, the rotor electromagnet's north pole is passing L2. In the first generator, the L1 coil will be getting maximum current (voltage wave is at the peak of the sine wave) but the voltage wave will be at the beginning of the negative portion of the sine wave in the second generator. But the two generators are joined with 275 kV wires, therefore there will be a huge explosion due to possibly 300 kV difference between the wires that are joined. This will cause the smaller generator to be blown up. This is why the protocol for generators is that if the frequency of one generator is even slightly different from the others, the protection system will automatically cut it off from the grid and without enough power supplying the demand, the whole grid will trip [15].

To solve this sudden supply and demand inequality, assuming all generators are ICE, they press the accelerator 30 % extra as in the Sarawak Grid (the spinning reserve is

30 % of the grid's output). Therefore, if one generator in the grid suddenly trips, extra DC is sent to the rotor of other generators and immediately extra current comes out of the stator coils of all the other generators. But this will make the stator coils in all the other generators stronger electromagnets also. But because the accelerator is already being pressed 30 % extra, there is enough mechanical power to overcome the extra magnetic attraction between the stator and rotor coils. Therefore, all generators can still turn at 50 Hz (60 Hz in the USA). But if each householder has BEV or many plugged into the grid, the grid can take power from people's cars and the Power Utility need not press all the accelerator in all ICE engines 30 % extra all the time [15]. That is an immediate reduction of 30 % climate changing pollution while all other attempts at reducing pollution offer only 1-2 % improvement.

This author studied in SD, USA in the late 1980s and early 1990s and there was an electric plug point at every car park in SD, USA. That was much before the popularization of BEV by Tesla, Inc. The reason why SD needed an electric plug point at every parking spot is because it can get too cold during the deep winters and plug point will power the heater in the car to prevent the fuel and engine oil from freezing. Therefore, humanity has already long developed methods to place an electric plug point in all car parks much before the 1980s. With a plug point in each car parking spot, even if a generator within the grid trips during working hours, the grid can take power from people's cars which are parked at their workplaces [15].

It must be noted that financial buildings like banks have three power sources, the main supply from the grid, a battery bank and an ICE powered generator. This is because only the battery can immediately cut in upon a grid blackout and keep computers (and therefore financial information) alive. After a while the standby generator will kick in and supply the banks and also recharge the backup batteries. Therefore, battery power output from BEVs will be the fastest backup for the grid.

This is why India, China, Indonesia, Britain, France, and numerous countries have already specified a cutoff date for all vehicles to be BEV. The majority of countries have legislated bans on ICE cars by 2035. Meaning, though there is a severe lack of engineers in politics worldwide, even non-engineer politicians can understand that BEVs can immediately reduce pollution and stabilize the grid [15].

2. Literature Review and current data

William Morrison of the USA developed the first

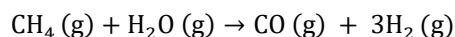
practical electrical vehicle in 1890. People who observed it at the time appreciated that the vehicle did not emit smelly gasoline and exhaust fumes and did not need to be cranked to start [16]. But the invention of the car electric starter in 1912 made ICE engines more attractive for consumers mainly because of the short time to refill gas compared to charging batteries [17]. The vast reserves of oil were coincidentally discovered at that time. By the mid-1920s EVs totally lost market share [18]. It was the 1973 oil crisis that got humanity to once again revive EVs. Hybrids were widely sold and appreciated in the later portions of the 20th century [19]. Tesla, Inc played a pivotal role in reintroducing BEV by first getting humanity away from the concept that BEVs are weak vehicles suitable mostly as golf course carts. To achieve this, Tesla Inc. made premium and expensive cars for the rich which depicted that BEVs can be powerful cars. Then many car races of high-end ICE sports cars versus simple BEVs moved humanities' perception away from the weakness of BEVs.

Tesla, Inc introduced BEVs together with rooftop solar panels which was viewed favorably by people worldwide even though those outside the USA did not experience Tesla's business as much. It was that vision that changed humanity's perception of the practicality of BEV. The final anxiety of driving a BEV is being solved as charging infrastructures are being expanded worldwide [20].

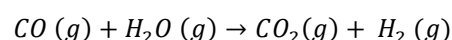
There is a very common argument that BEV still uses coal in the power station to produce electricity, but the fact is that even if coal is used to generate electricity, the best pollution control devices can be installed in their chimneys since they have economies of scale. Installing similar pollution controls in each ICE car will make them too expensive for most buyers. In Sarawak, Malaysia for example, there are three coal electricity generating plants namely Sejinkat, Mukah and Belingian, arranged in accordance with the age of the plants. But Belingian has the best pollution control equipment, Mukah has less and Sejinkat has even less.

Typical pollution control in coal plants includes Scrubber Systems to remove SO_x and PM (particulate matter). There are also EPSs (electrostatic precipitators which use electric fields to trap ions and therefore capture PM. Then there is SNCR (selective non-catalytic reduction) where ammonia or urea is injected into the exhaust stream to reduce NO_x emission. Sound waves are a new concept which can be further developed to control pollution though it is at an infancy today. Powerful sound generators positioned at strategic points in the chimney will be required. Currently they are not developed enough to be well utilized in coal power plants [21].

FCEV requires H₂. The SMR (steam methane reforming) process is the primary method of deriving H₂. It produces 95% of H₂ in use today. Natural gas (which is 70 – 90 % methane or CH₄). Initially the natural gas is desulfurized. The desulfurized gas is mixed with water boiled to 700-1000°C in a furnace. Then Ni catalyst is used to achieve the following reaction [22].



Therefore, the products are a mixture of H₂ and carbon monoxide (CO) and some unreacted methane steam. CO is a pollutant, but it can be utilized to derive even more H₂ using the process below:



This is an exothermic reaction which utilizes the catalyst of CuO or Fe₂O₃. The eventual mixture still contains H₂, CH₄ and CO₂ plus other trace gasses. A process called pressure swing absorption (PSA) separates the gasses based on their different absorption properties. Membranes are also used to allow H₂ to pass through and block the other gasses. The final product is a high purity H₂ which is currently mainly used in the manufacture of ammonia for fertilizer and for petroleum refining. The downside is that the whole process described above utilizes a significant amount of heat and releases a significant amount of CO₂ unless Carbon Capture and Storage (CSS) is utilized where the CO₂ is stored underground mainly to build up pressure in O & G wells to push out more fossil fuels. But of course, in places where there are no O & G wells this is not possible [23].

The other process to derive H₂ is via electrolysis of H₂O which needs 55 kWh of electricity to produce 1 kg of H₂. To get a sense of how much 55000 Wh is, an incandescent light bulb which is switched on for an hour uses 50 Wh. A refrigerator uses 500 Wh if it is switched on for an hour. A toaster which is switched on for an hour uses 700 Wh. Therefore with 55000 Wh a toaster can be switched on for [24]:

$$\frac{55000}{700} = 78 \text{ hours}$$

And that is the power to produce 1 kg of H₂ which can provide a H₂ car 600 km range. A typical BEV achieves about 6.4 km per kWh. Therefore 55 kWh can provide a range of:

$$55 \text{ kWh} \times 6.4 \text{ km} = 352 \text{ km}$$

But BEV battery capacity has increased as shown in Figure 1, and cost has come down. From 2008 to 2021, battery cost has decreased by 87 % [25].

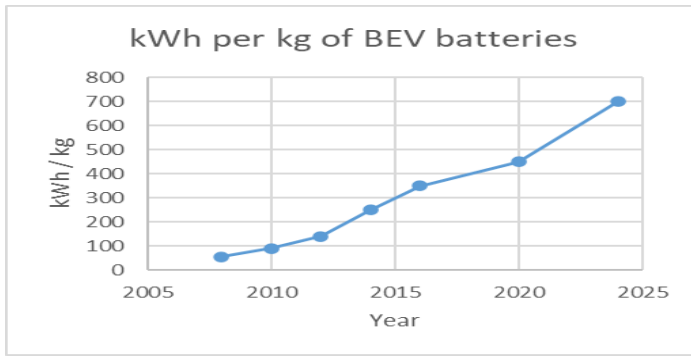


Figure 1: Battery capacity increase, kWh /kg for BEV batteries trend

BEVs have also saved owners thousands of dollars per year because electricity is cheaper than fuel or hydrogen. BEVs can convert 80% of electrical power from the grid to power the wheels while an ICE only converts about 20 % of the energy of fuel to kinetic energy of the car. And H₂ sees energy losses in hydrogen production, storage (compression cost) and then there are losses as it generates electricity in fuel cells which is DC and there are losses as this DC is converted into AC because most FCEV uses AC motors. Using DC motors is not practical because DC motors use carbon brushes which will have to fail one day. Commutator less DC motors where sensors detect the position of the rotor and flip the polarity of DC going into the rotor are to date of small sizes.

Kuching, Malaysia is using hydrogen buses. The expert in that setup informed this author that it takes 1 MW of electricity to generate 20 kg of H₂, which is similar to the previous number of 55,000 kWh.

$$\begin{aligned}
 20\text{kg} &\rightarrow 1\text{MW} \\
 1\text{kg} &\rightarrow \frac{1}{20} = 0.05\text{MWh} = 50\text{ kWh} \\
 \therefore 1\text{ kg} &= \\
 @ \$0.25 \text{ per kWh (USA average)} &\equiv \\
 50\text{kWh} &= 50,000 \times 0.25 = \$12,500
 \end{aligned}$$

The H₂ is stored in a 1200 kg tank where 50 kg escapes daily because hydrogen is the smallest atom in the universe. Therefore, the daily loss due to H₂ escaping from the tank is \$625,000.

$$kg = 50\text{ kg} \times \$12,500 = \$625,000$$

It should be noted that H₂ is normally placed in a tank in H₂ cars at 700 atm. It takes more energy to compress H₂ to 700 atm than the energy it provides to FCEV cars for transportation. That is not counting the high energy and CO₂ pollution to derive H₂ from natural gas. Humanity is living at 1 atm. A vehicle collision that punctures a 700 atm tank will be disastrous for a lot of humans near the punctured tank.

Countries like Japan and India were initially very interested in H₂ because they can finally get out of the

control of O & G producing countries since H₂O is all around Japan and India. They have therefore invested quite a significant amount of money in producing H₂, but it will not go to waste [26]. Large combustion engines can be run with H₂. The bigger the combustion engine the more efficient it is because there is more space for the air and fuel to mix. Using H₂ to run fuel cells is 200 % more efficient than burning them in a combustion engine [27]. But the problem with the fuel cell's output is that it is DC which must be converted to AC so that it can be stepped up with a transformer. This must be done to reduce the current in the wires. For example, the main grid wires of 275 kV in Sarawak, Malaysia have current within each cable of which range from 10 A to 59 A while the current coming out of a small car battery upon starting can reach 110 A. Of the four parameters of electric power of P, V, I, R, wire size only depends on I. If transformers are not used, the wires in the main grid will have to be up to 2 m in circumference [28].

A transformer can only step up or down AC. Basically, a transformer can be compared to using a magnet to pass over a wire to generate electricity. If a hand is shaped according to Fleming's Right-Hand Rule to pass a wire, what the wire will see is a few magnetic field lines when the magnet is far away from the wire, and as the magnet is moved closer to wire, the number of magnetic field lines increases till it reaches a maximum when the magnet is above the wire and then decreases as the magnet moves away from the wire. Basically, the number of magnetic field lines must always change to generate a current in the wire below the magnet. Even if the strongest magnet in the world is placed above a wire, there will be lots of magnetic field lines, but the number of field lines is not changing so there will be no current generated in the wire below it. Therefore, the Fleming Right-Hand Rule with the three fingers is normally defined as F for force or direction of the movement of the magnet, B for the direction of magnetic field, and I for current generated. The F can be changed from Force to varying Field density (still using F). In a transformer, there is no movement, but the AC flowing in the transformer's primary coil already has a varying Field density because in the far away generator, the rotor moves, passing over the stator coils of the generator. [28]

Therefore, the movement of a magnet across a wire to generate current in the wire is replaced by the movement of the rotor electromagnet across the stator coil wire in a faraway power station. Therefore, as the voltage goes up and down following the sine waveform, there is a varying Field density experienced by the transformer's secondary coil. The primary and secondary coil is not connected with wires, an iron core which is the best conductor of

magnetic field lines (high permeability) moves the field lines from the primary coil to the secondary coil. Note that current is a measure of magnetic field strength of a current carrying wire while voltage is a measure of the electric field strength of a current carrying conductor [28].

If fuel cells are used, the cost of the DC to AC inverters must be considered. It is currently quite highly priced. In fact, the former Energy Secretary of the US, Steven Chu wanted the main grid of the USA to be DC but the cost of the inverters to convert AC to DC was a deterrent [29].

There is also the possibility of using H₂ to run GTs. Note the GT which is designed to combust natural gas cannot combust H₂ directly. Some modifications must be made but GE Venova and Mitsubishi Heavy Industries have already built GTs that can run with H₂. It must be noted that one of the biggest problems with H₂ is that it causes embrittlement in metals because it is the smallest atom in the universe and can get into metals and make them brittle. But the insides of GTs are increasingly being changed into ceramics. The ceramics are in the form of tile coatings on metallic components to withstand the high combustion temperature [30].

The cost of building the EV infrastructure is very much cheaper than building fuel stations. A gas station in the US costs between \$200,000 to \$2,000,000 (taking a median of \$1,100,000). But an EV charging station costs \$1,676 each (22 kW). Therefore, for the price of one fuel station the number of EV chargers that can be built is [31]:

$$\frac{1,100,000}{1676} = 656 \text{ EV charging stations}$$

This is why as of 2024, a single company, Tesla, Inc, has already built 6,000 Supercharger stations in the world (2,300 in N. America, 2,400 in Asia Pacific and 1,100 in Europe). The total number of supercharger stations in all these 6,000 stations is 55,000 [32].

EV batteries are also recyclable with two methods namely pyrometallurgy (high heat treatment) and hydrometallurgy (using chemical solutions) [33].

Therefore, converting the whole O & G plus ICE engine for transportation of humanity to electric vehicles is much more practical because the energy source can be changed anytime, and the infrastructure of EV can remain the same. For example, India is rooting for thorium energy which promises thousands of years of electricity for India given her reserves of thorium. And this can be used to power electric vehicles including planes.

The RMI or the RMI India Foundation (RMIEFI) which conducts research specifically on India's clean energy transition have predicted that by 2030 EV will form about 74 % of vehicle sales in India. It predicted that ICE cars

will peak by 2025 and will experience a free fall after that. By 2027 BEV will be cheaper than ICE cars [34].

The use of renewable energy must be coupled with Artificial Intelligence (AI). Solar panels for example are connected in series. If one panel in the series is covered by a shadow, all the rest in the series will be affected. Similarly in batteries, there are connections in series to get the required voltage and in parallel to get the required charge (or number of hours the battery can supply power). If one battery in a series fails, all batteries in that series will be affected. AI plus contactors at the positive and negative terminal plus one more contactor to jump over that faulty battery can be used to overcome this problem [35].

Toyota's CEO announced that he was leaving BEV. This is just to justify to his stakeholders that they can all keep their business. The underlying truth is that there are only 20 moving parts in an EV, and the combustion cars have about 30,000 parts which have been made from great grandfather to now with impressive innovation to reach perfection (as Toyota can do best) and all must shut down. That is the loss Toyota is trying to avoid; BEVs are a great disruptor. Of course, the O & G companies / countries will also pay them very well to take this narrative [36].

One more argument against BEV is that it utilizes lots of resources during manufacturing. But studies done have shown that the battery material for an average BEV is 170 kg. While an ICE will use up 17,000 L (12,580 kg) of fuel over its lifetime. If the fuel is filled into tanks, it would be as high as a 90 m building (diesel cars will make a 70 m building). Considering all the material dug from the ground to make a BEV versus an ICE, the cost of the material over its lifetime will be 350 times more for ICE [37].

One of the arguments against BEV is that it emits electromagnetic fields (EMF) which may affect humans. This is most probably a paid argument by the O & G and ICE industries to the medical establishment. The truth is that humans already live in a giant electromagnet. The outer core of the earth is mostly molten iron plus other heavy metals. The earth was initially a spinning ball of gas which was thrown out of the sun, most probably due to some collision of a body with the sun. The force of that collision caused the earth to be in its particular orbit. Initially as the gasses spun, the heavier particles will move out due to centrifugal force, but as more and more particles joined this ball of gas, it acquired the force gravity such that gravity exceeds the centrifugal force, therefore the heavier elements moved to the center of the ball. Therefore, iron and all the heaviest elements in the periodic table moved to the center and the lighter

elements like Si and the gasses moved out to the Mantle, Crust and to the atmosphere above earth. Hence the earth has a solid Fe core which is too hot to exhibit any magnetic properties. The outer core, however, has molten Fe and since Fe will always release its valence shell electrons, there are free electrons moving parallel to each other in the outer core. Therefore, just as electrons moving parallel to each in the coils of a solenoid turns it into a magnet, the outer core is a giant electromagnet which creates a force shield around the earth, protecting it from harmful e-m waves from outer space. Therefore, humanity has always lived in a giant electromagnet and need not be concerned by the EMF emitted from the induction motor of an electric vehicle [28].

There are many BEV owners who get much of the power for the car via solar panels. But renewable pathways must be trodden carefully. In the current state, renewal power is causing many problems. States like California and the country of Australia which went heavily into renewals are today facing a power crisis. Backing up a few kW of power due to sudden no wind situation is easy, but when it gets to a few gigawatts, it is nearly impossible. The batteries are not ready today. Only hydro can back it up. But this author believes that criteria is not considered in building large wind farms. A hydro powered generator can energize the grid in 20 seconds. But even a 20 s delay will require sudden tripping of lots of loads (load shedding) in the grid. For some loads like high-tech factories, 20 seconds is enough to damage all products they are making. For a high-tech plant like a chip making plant in Kuching, Malaysia, it takes one month to make one batch of products and that whole batch would be damaged with one power trip.

As mentioned in the introduction, this author was an engineer in the Grid Control Room of the 728 km wide Sarawak Grid. It was his job to call the different power stations to start or stop generators to ensure supply and demand was equal. Table 1 is the time it takes for different types of electric power engines to energize the grid upon getting instructions to start from the Control Room.

Table 1: The time to energize the grid for different types of electric power engines

Coal power station engines	8 hours
Gas Turbine engines	0.75 hour
ICE engines	1.25 hour
Hydroelectric engines	20 seconds

The main 275 kV grid wires were continuously observed, if the voltage goes above 275 kV, there is more supply than demand and if it goes below 275 kV, there is more demand than supply. Actions must be taken by starting or stopping generators according to the main grid lines voltage which is 275 kV. Another method used was to energize or de-energize large inductors, called reactors located in substations to bring down or bring up the main grid voltage respectively.

The 20 seconds for a hydroelectric powered generator engine is only if the penstock pipe already has water flowing within it. In this case, energizing a hydro generator requires putting DC into the rotor coils and immediately AC flows out of the stator coils. This span of time to start various generators is the reason why intermittent renewals are not a good idea. Also starting and stopping large ICE, coal or GT plants frequently will damage them. Even ICEs are just kept running for weeks at a time. This may be surprising for most car owners who never keep their cars running for weeks at a time.

Some point to the wind and solar in Western countries, but power projects are so huge that only politicians are high enough to decide on them. This author has seen that. He was in the audience when the energy minister of Malaysia announced she wanted the people of KL to buy 1 GW of solar panels and install it on their roof and supply to the grid. This author wrote a letter to her about the problems with that, which probably ended up in her rubbish bin. If there is a sudden cloud cover over KL as can happen in equatorial Malaysia, there is no way to back up 1 GW. But the Utility manager of KL is not going to lose his job over it. If 1 GW of solar power enters the grid, he is going to run 1 GW of coal as a backup. Therefore, the citizens of KL have to spend a huge amount of money to purchase solar panels and it is being paid for by the Utility as their bills are reduced but the Utility will actually be running 1 GW of coal for this whole political decision.

The only reason humanity is not driving electric cars as much is resistance to electric cars from the O & G plus ICE industries.

3. Maneuverability of BEV

Both the BEV and the FCEV use an induction motor as the prime move for the wheels. Therefore, the BEV is built like a skateboard with a flat base which is the battery compartment and four wheels. There are only two cylinders that run the whole car, one is the induction motor whose shaft is connected directly to the shaft of the rear wheels. The other cylinder is the Variable Frequency Drive (VFD). In between the two is a gearbox as shown in

Figure 2 [38].

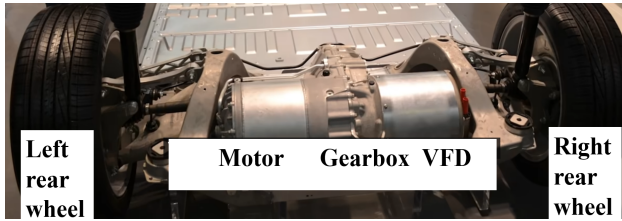


Figure 2: The parts between the rear wheels of a BEV

The VFD is basically a computer that has software that replaced many mechanical systems in the car. It makes the AC waves going into the induction motor narrower (smaller wavelength) to get the car to move faster and make the AC sine wider (bigger wavelength) to get the motor to go slower. If the amplitude of the sine wave is higher, the car will have more power. The gearbox is just a simple open differential. The differential was developed from the days of the horse cart. The four wheels of a horse cart are independent. But when an ICE was first used to drive a cart, the cart cannot turn because to turn, the outer diameter wheel needs to move faster. Therefore, a differential was invented to enable the outer diameter wheel to move faster than the inner diameter one. The gearbox of a BEV just reduces the speed two times from the speed of the induction motor; this also provides more power to the car. Fig. 3 and Fig. 4 shows that BEV is powerful at a wide range of speed while an ICE is only powerful at a particular speed.

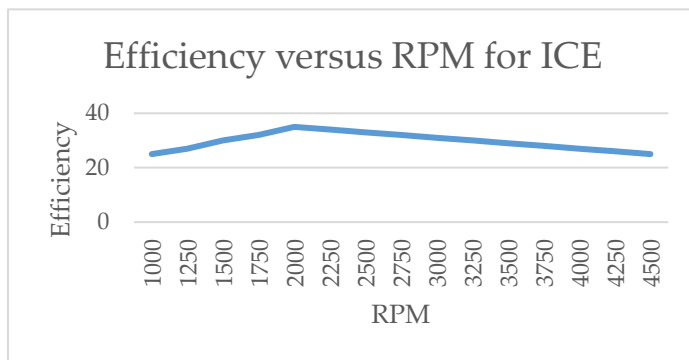


Figure 3: Efficiency versus RPM for ICE cars

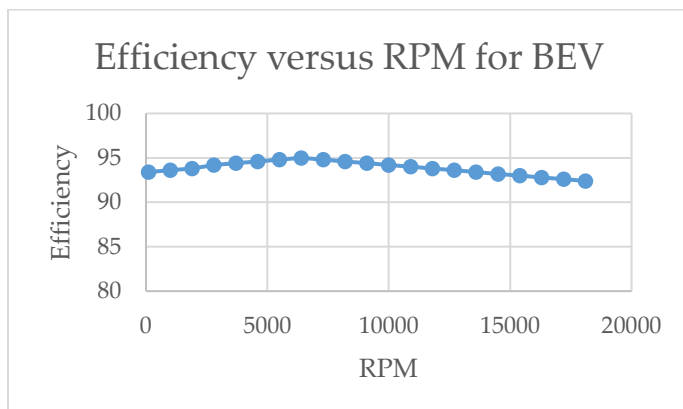
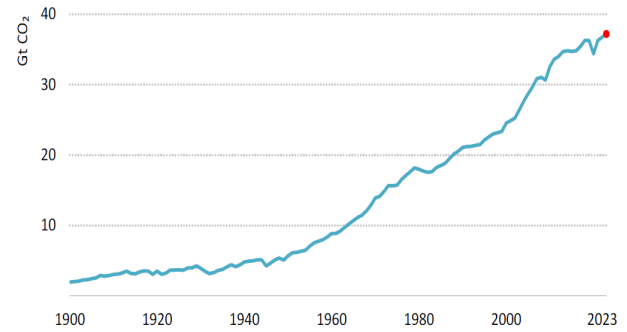


Figure 4: Efficiency versus RPM for BEV cars


Figure 5: Global CO₂ emission data from IEA (International Energy Agency) [39]

4. Conclusion

The conclusion of this work is that the argument against the BEV is just a fear of business loss from the O & G and ICE industries. The leaders of the old system, as for old kingdoms which still have much power, do not want to lose hold of their power. They could just use their huge financial resources to move into the BEV and other industries but their vested interest in the O & G and ICE is just too great. It is not just a change in technology, the mindset and expertise must change to get into BEV. They prefer to fight the introduction of BEV. BEV is capable of immediately clearing up pollution from the biggest cities of earth where 56 % of humanity lives. BEV is also very useful to keep the integrity of the electrical power grid including creating a large energy saving and pollution reduction (up to 30 %) for the electric power grid. It must be noted that the electric power grids of the earth are currently the biggest polluter of earth. The second biggest polluter is transportation (motorcycles, cars, trucks, trains and planes). Together these two industries produced 37 billion tonnes (Gt) of CO₂ in 2023. Power production emits 33 % of CO₂ pollution and the transportation industry emits 24 % of pollution emitted [40]. This paper explains how CO₂ pollution can be reduced from both these sectors with the widespread usage of BEV. Fig. 5 is a picture that tells a thousand words which depict what humanity needs to do.

Conflict of Interest

The authors declare no conflict of interest.

References

- [1] R. D. Reitz, "Directions in internal combustion engine research," *Combustion and Flame*, vol. 160, no. 1, 1-8, Jan. 2013, doi: 10.1016/j.combustflame.2012.11.002.
- [2] H. N. Gupta, *Fundamentals of Internal Combustion Engines*. PHI Learning Pvt. Ltd., 2012, ISBN: 978-81-203-4680-2.
- [3] M. S. Jneid, P. Harth, and P. Ficzer, "In-wheel-motor electric vehicles and their associated drivetrains," *International Journal for*

- Traffic and Transport Engineering*, vol. 10, no. 4, 415-431, 2020, doi: 10.1109/TIA.2015.2399617.
- [4] M. B. Neeraja, *Electric Vehicles*. Academic Guru Publishing House, 2023, ISBN: 978-81-967548-1-5.
 - [5] R. Pradhan, N. Keshmiri, and A. Emadi, "On-board chargers for high-voltage electric vehicle powertrains: Future trends and challenges," *IEEE Open Journal of Power Electronics*, 2023, doi: 10.1109/OJPEL.2023.3251992.
 - [6] W. M. Tsutsui, "W. Edwards Deming and the origins of quality control in Japan," *Journal of Japanese Studies*, vol. 22, no. 2, 295-325, 1996, doi: 10.2307/132975.
 - [7] J. D. Hunt, A. Nascimento, N. Nascimento, L. W. Vieira, and O. J. Romero, "Possible pathways for oil and gas companies in a sustainable future: From the perspective of a hydrogen economy," *Renewable and Sustainable Energy Reviews*, vol. 160, p. 112291, 2022, doi: 10.1016/j.rser.2022.112291.
 - [8] M. A. Raji, H. B. Oloodo, T. T. Oke, W. A. Addy, O. C. Ofole, and A. T. Oyewole, "Real-time data analytics in retail: A review of USA and global practices," *GSC Advanced Research and Reviews*, vol. 18, no. 3, 059-065, 2024, doi: 10.30574/gscarr.2024.18.3.0089.
 - [9] M. Amiti and S. Heise, "U.S. market concentration and import competition," *The Review of Economic Studies*, 2024, doi: 10.1093/restud/rdae045.
 - [10] R. Vezzoni, "How 'clean' is the hydrogen economy? Tracing the connections between hydrogen and fossil fuels," *Environmental Innovation and Societal Transitions*, vol. 50, p. 100817, 2024, doi: 10.1016/j.eist.2024.100817.
 - [11] M. J. Bradshaw and T. Boersma, *Natural Gas*. John Wiley & Sons, 2020, doi: 10.1111/1365-2745.14264.
 - [12] C. Fletcher, W. J. Ripple, T. Newsome, P. Barnard, K. Beamer, A. Behl, ... and M. Wilson, "Earth at risk: An urgent call to end the age of destruction and forge a just and sustainable future," *PNAS Nexus*, vol. 3, no. 4, p. pgae106, 2024, doi: 10.1093/pnasnexus/pgae106.
 - [13] P. Karunakaran, M. S. Osman, V. Karuppanna, S. C. Cheng, M. D. Lee, M. D. Fadhillah, and A. K. S. Lau, "Eddy Current versus Joule Heating Effects for a Cable Suspended in an Iron Pipe," *IEEE Xplore*, 2021, doi: 10.1109/INOCON50539.2020.9298438.
 - [14] P. Karunakaran, M. S. Osman, V. Karuppanna, S. C. Cheng, M. D. Lee, A. Richard, and A. K. S. Lau, "Electricity Transmission Under South China Sea by Suspending Cables Within Pipes," *2020 International Conference for Emerging Technology (INCET)*, Aug. 2020, doi: 10.1109/INCET49848.2020.9154119.
 - [15] P. Karunakaran, *Electrical Power Simplified*. AuthorHouse, 2018, ISBN: 978-1-5462-6246-6.
 - [16] R. Bhardwaj and S. Gupta, "Evolutionary progress of the electric car market with future directions," in *Latest Trends in Renewable Energy Technologies: Select Proceedings of NCRESE 2020*, Springer Singapore, 2021, 315-321, doi: 10.1007/978-981-16-1186-5_27.
 - [17] N. Burton, *History of Electric Cars*. The Crowood Press Ltd., 2013, ISBN: 978-1-84797-571-3.
 - [18] D. S. Painter, "Oil and the American Century," *Journal of American History*, vol. 99, no. 1, 24-39, Jun. 2012, doi: 10.1093/jahist/jas073.
 - [19] M. E. Staub, "Snake Oil and Gaslight: How the Petroleum Industry Got in Touch with Nature," *Environmental Humanities*, Duke University Press, vol. 15, no. 2, 85-104, 2023, doi: 10.1215/22011919-10422300.
 - [20] D. Ronanki, A. Kelkar, and S. S. Williamson, "Extreme fast charging technology—Prospects to enhance sustainable electric transportation," *Energies*, Electric Mobility and Transportation Innovation (E-MOTION) Laboratory, Smart Transportation Electrification and Energy Research (STEER) Group, vol. 12, no. 19, p. 3721, 2019, doi: 10.3390/en12193721.
 - [21] J. A. Gallego-Juárez, E. Riera-Franco De Sarabia, G. Rodríguez-Corral, T. L. Hoffmann, J. C. Gálvez-Moraleda, J. J. Rodríguez-Maroto, ... and M. Acha, "Application of acoustic agglomeration to reduce fine particle emissions from coal combustion plants," *Environmental Science & Technology*, vol. 33, no. 21, 3843-3849, 1999, doi: 10.1021/es990002n.
 - [22] M. Garg, D. Gera, A. Bansal, and A. Kumar, "Generation of electrical energy from sound energy," in *2015 International Conference on Signal Processing and Communication (ICSC)*, 2015, 410-412, doi: 10.1109/ICSPCom.2015.7150687.
 - [23] Y. Bai and M. F. Cotrufo, "Grassland soil carbon sequestration: Current understanding, challenges, and solutions," *Science*, vol. 377, no. 6606, 603-608, 2022, doi: 10.1126/science.abo238.
 - [24] A. Chuneekar and A. Sreenivas, "Towards an understanding of residential electricity consumption in India," *Building Research & Information*, vol. 47, no. 1, 75-90, 2019, doi: 10.1080/09613218.2018.1489476.
 - [25] S. Vengatesan, A. Jayakumar, and K. K. Sadasivuni, "FCEV vs. BEV—A short overview on identifying the key contributors to affordable & clean energy (SDG-7)," *Energy Strategy Reviews*, vol. 53, p. 101380, 2024, doi: 10.1016/j.esr.2024.101380.
 - [26] S. K. Kar, A. S. K. Sinha, S. Harichandan, R. Bansal, and M. S. Balathanigaimani, "Hydrogen economy in India: A status review," *Wiley Interdisciplinary Reviews: Energy and Environment*, vol. 12, no. 1, p. e459, 2023, doi: 10.1002/wene.459.
 - [27] Z. Stępień, "A comprehensive overview of hydrogen-fueled internal combustion engines: Achievements and future challenges," *Energies*, vol. 14, no. 20, p. 6504, 2021, doi: 10.3390/en14206504.
 - [28] P. Karunakaran, *Electrical Power Simplified*. UTS Publishers, 2023, 243 pages, ISBN: 978-629-98726-2-7.
 - [29] E. Hand, "Newsmaker of the year: The power player," *Nature*, vol. 462, 978-983, 2009, doi: 10.1038/462978a.
 - [30] P. Chiesa, G. Lozza, and L. Mazzocchi, "Using hydrogen as gas turbine fuel," *Journal of Engineering for Gas Turbines and Power*, vol. 127, no. 1, 73-80, 2005, doi: 10.1115/1.1787513.
 - [31] A. C. Lusk, X. Li, and Q. Liu, "If the Government Pays for Full Home-Charger Installation, Would Affordable-Housing and Middle-Income Residents Buy Electric Vehicles?," *Sustainability*, vol. 15, no. 5, p. 4436, 2023, doi: 10.3390/su15054436.
 - [32] Y. Babar and G. Burtch, "Recharging Retail: Estimating Consumer Demand Spillovers from Electric Vehicle Charging Stations," *Manufacturing & Service Operations Management*, 2024, doi: 10.1287/msom.2022.0519.
 - [33] Y. Kotak, C. Marchante Fernández, L. Canals Casals, B. S. Kotak, D. Koch, C. Geisbauer, and H. G. Schweiger, "End of electric vehicle batteries: Reuse vs. recycle," *Energies*, vol. 14, no. 8, p. 2217, 2021, doi: 10.3390/en14082217.
 - [34] A. B. Styczynski, "Business Model Innovations for Electric Vehicle Adoption in India: An Ecosystem Perspective," in *India's Energy Revolution*, Routledge India, 2024, 190-213, ISBN: 9781003281818.
 - [35] P. Karunakaran, P. Karunakaran, S. Karunakaran, A. Karunakaran, F. Cassidy, V. Karuppan, and S. Haridas, "The Optimization of Solar Photovoltaic System for Rural Off-grid Villages," *IEEE Xplore*, 2022, doi: 10.1109/ICONAT53423.2022.9725993.
 - [36] D. Keohane, et al., "'Told you so' moment for Toyota on hybrids. Carmaker enjoys measure of vindication after warning repeatedly that consumers would balk at going full electric," *Financial Times*, p. 9, 27 Feb. 2024. [Online]. Available: link.gale.com/apps/doc/A784128571/AONE?u=anon~7503423a&sid=googleScholar&xid=9740b1f3. Accessed: 26 June 2024.
 - [37] A. F. Burke, J. Zhao, and L. M. Fulton, "Projections of the costs of light-duty battery-electric and fuel cell vehicles (2020–2040) and

- related economic issues," *Research in Transportation Economics*, vol. 105, p. 101440, 2024, doi: 10.1016/j.retrec.2024.101440.
- [38] M. Kumar, K. P. Panda, R. T. Naayagi, R. Thakur, and G. Panda, "Comprehensive Review of Electric Vehicle Technology and Its Impacts: Detailed Investigation of Charging Infrastructure, Power Management, and Control Techniques," *Applied Sciences*, vol. 13, no. 15, p. 8919, 2023, doi: 10.3390/app13158919.
- [39] G. Carrington and J. Stephenson, "The politics of energy scenarios: Are International Energy Agency and other conservative projections hampering the renewable energy transition?," *Energy Research & Social Science*, vol. 46, 103-113, 2018, doi: 10.1016/j.erss.2018.07.011.
- [40] S. Paul and R. N. Bhattacharya, "CO₂ emission from energy use in India: a decomposition analysis," *Energy Policy*, vol. 32, no. 5, 585-593, 2004, doi: 10.1016/S0301-4215(02)00311-7.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



PRASHOBH KARUNAKARAN

has done his bachelor's and master's degree from South Dakota State University, SD, USA in 1991 and 1993 respectively. He has completed his PhD Universiti Malaysia Sarawak 543 (UNIMAS) in 2012. Prashobh Karunakaran has

so far published 48 journal papers, presented in 31 Conferences and published 40 books and have received 13 awards. He worked as an engineer in an electric power utility and a computer hard disk manufacturing factory (WD) for 20 years before moving back to academia for a total of 10 years.

A Thorough Examination of the Importance of Machine Learning and Deep Learning Methodologies in the Realm of Cybersecurity: An Exhaustive Analysis

Ramsha Khalid ^{*1,2}, Muhammad Naqi Raza ¹

¹ Electrical Engineering Technology, University of Sialkot, Sialkot, 51310, Pakistan

² Electrical Engineering, University of Lahore, Lahore, 53720, Pakistan

*Corresponding author: Ramsha Khalid, University of Sialkot, Sialkot, Email: ramshakhalid2404@gmail.com

ABSTRACT: In today's digital age, individuals extensively engage with virtual environments hosting a plethora of public and private services alongside social platforms. As a consequence, safeguarding these environments from potential cyber threats such as data breaches and system disruptions becomes paramount. Cybersecurity encompasses a suite of technical, organizational, and managerial measures aimed at thwarting unauthorized access or misuse of electronic information and communication systems. Its objectives include ensuring operational continuity, safeguarding the confidentiality and integrity of sensitive data, and shielding consumers from various forms of cyber intrusions. This paper delves into the realm of cybersecurity practices devised to fortify computer systems against diverse threats including hacking and data breaches. It examines the pivotal role of artificial intelligence within this domain, offering insights into the utilization of machine learning and deep learning techniques. Moreover, it synthesizes key findings from relevant literature exploring the efficacy and impact of these advanced methodologies in cybersecurity. Findings underscore the substantial contributions of machine learning and deep learning techniques in fortifying computer systems against unauthorized access and mitigating the risks posed by malicious software. These methodologies facilitate proactive measures by predicting and comprehending the behavioral patterns and traffic associated with potential cyber threats.

KEYWORDS: Cybersecurity, Cyber attackers, Artificial intelligence, Machine learning, Deep learning, Communication systems, Unauthorized entry

1. Introduction

The Internet has transformed various facets of contemporary life, creating a global village where knowledge exchange and cultural interactions flourish. Networks form the backbone of this digital landscape, connecting devices such as computers and mobile phones. Their significance lies in enabling access to the Internet; without it, these devices lose much of their functionality. Through networks, data, information, and applications flow seamlessly via physical cables and wireless radio waves. Protecting personal data, crucial for transactions conducted over networks, is paramount amidst the looming threat of hackers aiming for identity theft or fraudulent activities [1]. The COVID-19 pandemic catalyzed a shift towards digital transactions with

minimal physical contact to mitigate virus transmission. This transition propelled widespread adoption of electronic transactions by institutions and businesses, highlighting their efficiency and accessibility benefits for consumers. Simultaneously, online shopping platforms, including those on social media like Facebook, witnessed increased activity, facilitating the sale and distribution of goods [2,3]. Moreover, educational institutions transitioned to online platforms for delivering education and training, reflecting a growing acceptance of digital learning modalities. Similarly, remote work gained traction as a viable option for both public and private sector organizations, facilitated by the widespread availability of Internet connectivity [4,5].

As remote work liberates employees from fixed locations, the sharing of online workspaces necessitates information security specialists to evaluate the associated business risks and prevent unauthorized access or hacking attempts from external parties [6,7]. Despite the implementation of sophisticated technical security measures by organizations to combat cyber threats, the human element remains a critical consideration, as employees' skills often constitute the weakest link in the security chain. It is imperative for employees to be vigilant against potential hacking or malicious software threats that may compromise their data unbeknownst to them. In addition to conducting awareness-raising activities such as training sessions and workshops for employees lacking expertise in cybersecurity, organizations must also enforce a range of technical safeguards. Practices posing threats to information security include leaving computers unlocked while unattended, abandoning devices in public settings, and disregarding company policies regarding password security. Consequently, there is a pressing need for further investigation into the risks associated with remote work.

1.1. Artificial Intelligence

Artificial intelligence techniques have emerged as some of the most advanced and invaluable tools in various fields, including cybersecurity and information security [8,9]. AI encompasses the capability of machines, electronic devices, software, applications, and gaming consoles to mimic human brain functions, such as awareness, memory, and data utilization, in decision-making processes [10]. Equipped with electronic brains, AI-enabled devices can analyze data and perform required operations, leveraging insights garnered from experimental data. The term "cybersecurity" has gained prominence in response to the widespread adoption and accessibility of Internet networks, particularly with the advent of 5G technology [11]. The proliferation of electronic crimes targeting data, information, and applications on computers and electronic devices underscores the imperative for robust security measures. Consequently, companies are increasingly turning to AI-based techniques to forecast cybercrime activities, preempt attacks, and thwart unauthorized intrusions into computer systems. Compared to human specialists, AI techniques offer superior efficacy in scrutinizing network users for authorization, thereby enhancing security protocols [12,13]. Moreover, their capacity for rapid learning, retention, and task execution translates into significant time and resource savings for experts. Notably, AI techniques excel in recognizing repetitive patterns, a feature invaluable in cybersecurity for identifying and analyzing user behaviors and predicting anomalous activities indicative of malware infiltration [14,15].

This paper makes a substantial contribution by elucidating the pivotal role of machine learning and deep

learning techniques in cybersecurity. It showcases their efficacy in mitigating intrusions and attacks on computer systems while elucidating their diverse applications within the cyber domain. Additionally, the paper provides a succinct overview of seminal studies leveraging these techniques in cybersecurity, scrutinizing their findings and elucidating their impact on decision-making processes. The data utilized in this paper are sourced from reputable news outlets and scholarly literature, streamlining the research process for cybersecurity scholars and practitioners.

The subsequent sections of this paper are structured as follows: Section 2 provides an examination of prevailing cybersecurity practices and the attendant challenges confronting computer systems. In Section 3, an overview of prominent datasets employed in attack and intrusion detection is presented. Sections 4 and 5 delve into the importance of machine learning and deep learning methodologies in cybersecurity, along with a comprehensive review of key literature utilizing these techniques. Finally, Section 6 offers concluding remarks.

2. Cybersecurity Practices

In recent years, the electronics and technology industry has experienced significant growth, becoming an integral part of daily life for individuals, indispensable for the fulfillment of business endeavors and projects. The functionality of modern devices relies on a suite of applications designed to serve human needs, necessitating comprehensive protective measures to safeguard against intrusions, hacking, attacks, and unauthorized access [16,17]. Concerns regarding hacking and data theft loom large for numerous companies and institutions. As organizations across diverse sectors increasingly acknowledge the paramount importance of their data, attention to cybersecurity has surged. This encompasses various facets, including measures to secure communication systems, data, and raw information, as well as virtual and physical components associated with operating systems. Secure applications, accessible only to authorized personnel, are vital elements within this framework [18–20]. Described as a combination of tools and practices, cybersecurity serves to defend computer systems' contents and thwart the infiltration of malicious software [21]. Table 1 provides a comprehensive summary of the various types of cybersecurity and their roles in safeguarding computer systems. Fundamental to cybersecurity are three key features: confidentiality, ensuring that unauthorized individuals cannot access or manipulate data within a computer system; integrity, preventing unauthorized modification or deletion of data; and availability, ensuring that data, information, and communications reach intended recipients without interception or decryption by unauthorized parties. Regardless of location, cyberattacks pose significant risks

to organizations, their employees, and their clientele, potentially resulting in profound consequences. Thus, it is imperative for employees to possess awareness of their organizations' cybersecurity protocols and adopt practices to mitigate associated risks. Table 2 illustrates significant instances of cyberattacks.

Table 1: Types of Cybersecurity and Their Functions

Cybersecurity Category	Functions
Application Security	Execute intricate codes to safeguard and encrypt data effectively [22]
Information Security	Ensure data protection from unauthorized access and alterations [23]
Infrastructure Security	Secure critical infrastructures such as power networks and data centers, ensuring absence of vulnerabilities [24]
Network Security	Protect networks from intrusions using tools like remote access management, two-factor authentication (2FA), and robust firewalls [25]
User Education	Provide valuable training sessions and conferences for employees and cybersecurity professionals [26]

Table 2: Types of Cybersecurity Attacks

Type	Description
Malware	A collection of malicious applications designed to damage systems and steal data [27]
Ransomware	Malicious software that encrypts data, disables systems, and restricts authorized user access [28]
Phishing	A common form of social engineering wherein individuals are manipulated into divulging sensitive information, posing significant security risks [29]
DDoS	Denial-of-Service attacks that disrupt systems, preventing user access to network resources, and inflicting financial or reputational harm on organizations [30]
SQL Injection	Exploits web security vulnerabilities to access, steal, modify, or delete data from websites, leading to system dysfunction [31]
Zero-Day Exploit	Newly discovered security vulnerabilities exploited by hackers

	to target computer systems, often leaving administrators with insufficient time to address the issue [32]
DNS Tunnelling	A sophisticated attack technique involving the encoding of system data and applications, challenging to detect [33]
XSS Attacks	Injection of malware into trusted websites, camouflaged as benign browser scripts [34]

The initial instance of malicious software, Creeper, surfaced in the 1970s with the capacity to destruct computer data. Upon infection, affected computers displayed a notable message on their screens: "I'm a creeper, catch me if you can!". In response to this emerging threat, the inaugural antivirus program, known as Reaper, was developed.

In 1903, Nevil Maskelyne made history as the first documented hacker by intercepting the inaugural wireless telegraph transmission, thereby revealing vulnerabilities inherent in Marconi's system. Concurrently, John Draper emerged as the pioneering cybercriminal, having discovered that the whistle included in Cap'n Crunch cereal boxes emitted a tone capable of deceiving telephone exchange signals, enabling him to place unauthorized free calls..

2.1. Cybersecurity Data Science

Data science encompasses the analysis of diverse data domains, ranging from life sciences to consumer behavior and cybersecurity. It plays a pivotal role in shaping the future of systems and cybersecurity fields, given its reliance on comprehensive data sets. Detecting cyber threats hinges on the meticulous analysis of security data, including files, records, and user activities within a network. Cybersecurity professionals leverage various techniques such as file hashes and custom rules, such as signatures or heuristics, to trace the origins of incoming data streams. While these manual methods offer unique advantages, they demand significant efforts to stay abreast of evolving threats and breaches.

Figure 1 illustrates the transformation of big data into actionable decisions, while Table 3 delineates various types of cybersecurity attacks. Data science endeavors to revolutionize information technology by leveraging machine learning and deep learning techniques to identify and address system vulnerabilities through feature extraction and pattern recognition from training data. Over the past decade, cybersecurity has increasingly relied on data science and artificial intelligence due to their capability to convert raw data into actionable insights and fortify system security.

In essence, data science offers an efficient approach to decision-making through tasks such as data engineering for data accumulation and analysis, data volume reduction through critical data filtering, discovery of unique patterns and data learning techniques, development of innovative data-driven security models, knowledge generation for mitigating false alerts, and optimization of system resources.

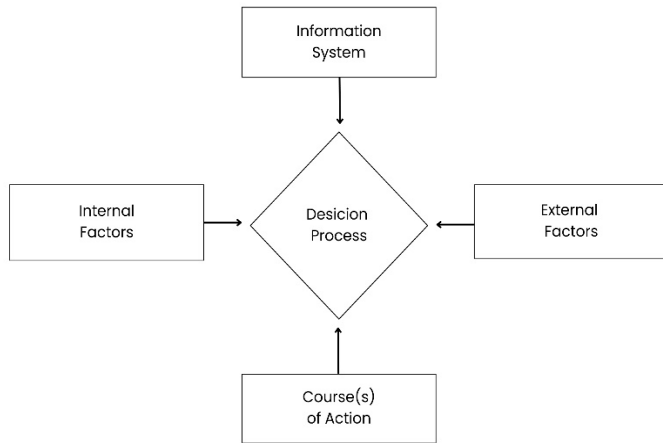


Figure 1: Representation of the process of data analysis and decision-making

Table 3: Prominent datasets utilized in cybersecurity research.

Dataset	Description
DARPA	Contains intrusion detection data, including LLDOS-1.0 and LLDOS-2.0.2, depicting connections between source and destination IP addresses, categorized by MIT Lincoln Laboratory for evaluating attacks and intrusion detection [35].
CAIDA	Encompasses distributed denial of service (DDoS) attack traffic and regular traffic traces, including unspecified traffic from a 2007 DDoS attack, facilitating evaluation of machine-learning-based detection models for identifying DoS activity on the Internet [36].
CTU-13	Comprises botnet traffic captured by a Czech university in 2011, featuring real botnet traffic mixed with normal and background traffic across 13 scenarios, suitable for data-based malware analysis employing machine learning techniques [37].

KDD'99 Cup	Widely used dataset since 1999 with 41 features for evaluating anomaly detection, categorizing attacks into probing, remote-to-local (R2L), user-to-remote (U2R), and DoS, suitable for assessing machine-learning-based attack detection models [38].
NSL-KDD	Revised version of KDD'99 Cup dataset, eliminating redundant records and addressing inherent issues to avoid bias towards frequent records [39].
MAWI	Dataset aiding researchers in anomaly detection, sourced from Japanese network research institutions, featuring traffic deviation labels in the MAWI archive, revised daily to include all traffic from applications and malware [40].
ISCX'12	Produced by Canadian Institute for Cybersecurity, containing 19 features for machine-learning-based attack detection and network penetration models, used in real-time with expert input to prevent system destruction and data theft [41].
Bot-IoT	Simulated Internet of Things environment dataset from UNSW Canberra Cyber Range Lab, featuring reliable traffic and various attack types, including DDoS, DoS, OS and service scan, keylogging, and data exfiltration, organized by protocol [42].
ISOT'10	Mix of malicious and non-malicious data traffic dataset from University of Victoria's ISOT research, utilized for evaluating models, machine-learning-based classification, and attack and penetration localization [43].
UNSW-NB15	Created using IXIA PerfectStorm tool in UNSW Canberra Cyber Range Lab, comprising contemporary synthetic attack activities and behaviors, with 49 features and 9 attack types, generated from TCPDUMP, ARGUS, and Bro-IDS tools [44].

2.2. Machine Learning in Cybersecurity

Electronic devices are experiencing rapid advancements, attracting a substantial following across various domains. However, the extensive communication and data exchange among these devices pose significant risks, notably concerning data breaches. Scholars advocate for leveraging machine learning techniques to

mitigate electronic threats, although these methods are still in developmental stages. Cyber threats are dynamic, necessitating adaptive solutions. Machine learning stands out as a potent tool due to its adaptability and learning capabilities. While effective in detecting and thwarting known malware attacks, machine learning faces challenges against novel threats. Operating within the realm of artificial intelligence, machine learning employs statistical operations to analyze data, extract insights, and aid decision-making. Its core objective is to enable computers to learn from expert-provided data [45–47]. Machine learning techniques encompass various rules and methodologies aimed at identifying or predicting novel data patterns or behaviors. These techniques find application in cybersecurity and can be categorized into supervised and unsupervised methods. Despite the substantial adoption of machine learning in cybersecurity, these tools remain imperfect, demanding significant human oversight, and necessitating continuous retraining of algorithms due to the inability to fully automate data processes [48,49]. This segment explores the functionality of machine learning techniques and their integration into cybersecurity practices. Figure 2 elucidates the operational framework of these techniques in detecting anomalies within a system.

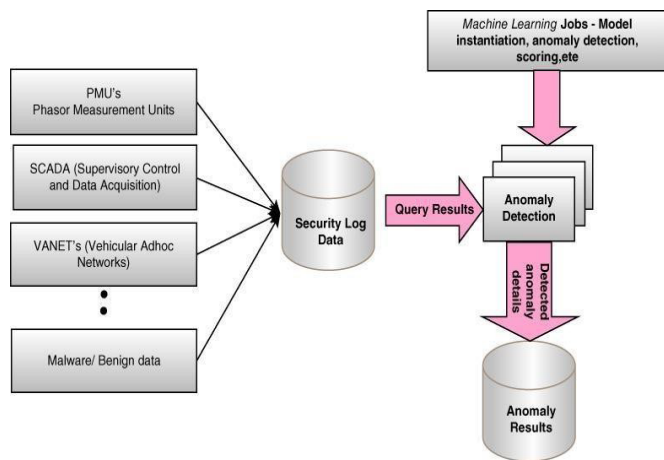


Figure 2: Utilizing machine learning techniques for anomaly detection [54].

2.2.1. Supervised Learning

Supervised learning functions methodically by establishing precise objectives and achieving them through a defined set of inputs [50]. Widely employed across various domains, supervised learning techniques offer straightforward implementation and monitoring. They are typically classified into classification and regression methods, which respectively categorize security data or anticipate specific security issues to be addressed in the future. The failure of an organization to avert an anticipated attack on its computer system can have far-reaching consequences, leading to substantial financial losses and necessitating a laborious recovery process in Figure 3. Consequently, the utilization of

machine learning techniques in cybersecurity is advocated to bolster data protection across all sectors and safeguard the data of both organizations and their users. Notable supervised learning techniques in classification include logistic regression, decision trees, support vector machines, k-nearest neighbors, and naive Bayes. These techniques are also applied in prediction tasks owing to their capacity to construct data-driven predictive models. For instance, the activities of users within a network, whether in a public or private institution, can be forecasted by continually tracking their actions, gathering relevant data, and discerning between processes initiated by human users and those generated by bots impacting the network. Meanwhile, prominent regression techniques in supervised learning, such as linear regression and support vector regression, are utilized to identify underlying causes of significant cybercrimes that profoundly affect individuals' lives and develop corresponding solutions. The distinction between classification and regression techniques lies in their respective outcomes: classification yields categorical or discrete results/effects, whereas regression produces numeric or continuous outputs/effects.

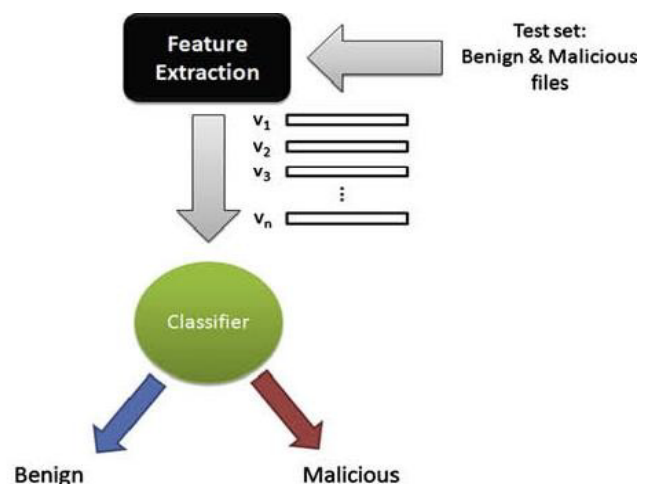


Figure 3: Utilizing machine learning techniques to classify unidentified datasets as either malicious or benign.

2.2.2. Unsupervised Learning

The primary objective of unsupervised learning techniques is to unveil patterns, structures, or insights from unlabelled data. However, within cybersecurity, malware often evade detection by dynamically altering their operations [51]. Clustering techniques, including k-means, k-medoids, and single linkage, constitute unsupervised learning methods aimed at uncovering hidden and intricate attack patterns and structures within large datasets. These techniques play a pivotal role in identifying and notifying users or developers about anomalies within systems, breaches of privacy policies, and unauthorized data accesses. Engineering tasks associated with these technologies, such as optimizing dataset features or extracting pertinent features related to specific security issues, are deemed essential for

conducting further analyses, irrespective of dataset scale. Moreover, security features are prioritized based on their significance. Additional methods like linear discriminant analysis, principal component analysis, non-negative matrix factorization, and Pearson correlation analysis contribute to addressing cybersecurity threats and uncovering clandestine programs using machine learning to preempt attacks and data breaches. In expert systems, rules are manually defined and implemented by a knowledge engineer in collaboration with a cybersecurity expert. Association rules learning aims to identify existing rules or relationships among datasets to extract relevant security attributes. Correlation analysis assesses the strength of relationships among datasets. Data mining techniques are categorized into frequent pattern-based, logic-based, and tree-based methods. Techniques such as AIS, Apriori, Apriori-TID, Apriori-Hybrid, FP-Tree, RARM, and Eclat are employed to formulate association rules capable of detecting intrusion and data theft issues. Table 4 enumerates the ten most impactful studies concerning the application of machine learning techniques in identifying attacks on operating systems.

Table 4: Scholarly works exploring the application of machine learning methodologies in identifying attacks and malicious software.

Article	Purpose	Techniques	Most Suitable Effect
[52]	Detect distributed denial-of-service (DDoS) attacks and malicious data.	MLP, K-NN, SVM, FL, ED, and MNB	The most suitable classification method is the MLP technique, achieving an F1-score of over 98% for emulated traffic and over 99% for real traffic.
[53]	Develop an intrusion detection system employing an efficient and reliable classifier.	SVM	The authors achieve a high accuracy of 98.62%, deemed excellent in intrusion detection.
[54]	Design a knowledge-based alert verification strategy with an intelligent filter to eliminate	KNN	KNN demonstrates the highest performance accuracy of 93.2% and the most robust F-measure of

	unwanted alarms.		91.8%.
[55]	Conduct experiments on five million Android applications to detect malicious software in the Android OS by recognizing features.	DL, FFC, Y-MLP, and DT	These methods achieve a peak performance accuracy of over 98% in identifying malicious software.
[56]	Detect ransomware tools (RANDS) operating within Windows environments through three stages (ransomware analysis, learning, and detection).	NB and DT	These methods attain an average classification accuracy of 96.27% in categorizing ransomware, with a 1.32% average real-time execution error.
[57]	Utilize a dataset of 300,000 attributes to predict and detect ransomware.	SVM	The technique reports an accuracy exceeding 88% in ransomware classification.
[58]	Monitor systems and detect intrusions by analyzing incoming data activities of servers and identifying malicious software.	DT with binary split	This technique achieves an impressive attack detection accuracy of over 99%.
[59]	Enhance the performance of the random forest strategy to detect misuse, anomaly, and hybrid-	New systematic frameworks of RF	These frameworks achieve a high detection rate in identifying and reporting anomalies.

	network-based intrusion detection systems (IDSs).		
[60]	Detect denial-of-service (DOS) attacks in software-defined networks (SDNs) and address cybersecurity management in SDN architectures.	KNN, RF, SVM, and ANN	These methods achieve an accuracy exceeding 98% in detecting DOS attacks.
[61]	Detect intrusions and identify malicious data through data mining techniques.	FCM, ANN, and SVM	These methods achieve a peak accuracy of 98.99% in detecting remote-to-local (R2L) attacks.

However, machine learning techniques are not without their constraints. For instance, they are incapable of identifying attacks that have not been previously encountered. Furthermore, the detection of behavioral patterns and anomalies may result in false positives if the behavioral restriction policy is overly broad. Conversely, implementing a stricter policy could diminish the effectiveness of these techniques. The selection of appropriate datasets is also crucial during the training phase of machine learning techniques in cybersecurity endeavors. Without proper training, these techniques may fail to deliver the anticipated results. When cyber adversaries become aware of a security system relying solely on a single defense technology, they may devise strategies to bypass such systems, such as through hacking. Nonetheless, a robust cybersecurity framework grounded in machine learning can leverage multiple complementary techniques.

2.3. Deep Learning in Cybersecurity

To address the intricate challenges in cybersecurity, various methodologies are employed based on specific criteria such as data volume, issue type, sensitivity, and decision tolerance. Deep learning techniques, leveraging parallel processing, prove highly effective in handling large-scale data and entail complex procedures [62–64]. This section critically examines the literature implementing deep learning methodologies for intrusion

detection, attack mitigation, and malware identification. These studies are succinctly outlined in Table 5. Deep learning architectures are not configured solely on local platforms; rather, they are deployed on server-based systems to ensure data integrity, confidentiality, and reliability, while also safeguarding against unauthorized access. The development of a robust deep learning model in cybersecurity involves two primary stages. Initially, the data transfer area is encrypted within the local environment before transmitting to the server. Subsequently, upon reaching the server, these encrypted data are processed, classified, and identified. For example, in character recognition from images, the first stage encompasses character encoding and transmission, while the second stage involves processing the received data and detecting any potential man-in-the-middle attacks between the server and local system, crucial for accurate data classification. This approach ensures secure information transmission to users, preventing unauthorized access to the system.

In networked environments, ensuring appropriate security measures and robust data preservation, storage, and transmission mechanisms are imperative responsibilities of the companies managing these systems. Breaches in networked systems vary depending on network activity and scale, with larger and more active networks encountering higher volumes of data requiring processing. To efficiently handle such data influx, parallel processing and deep learning techniques are favored for their speed and accuracy [65–67]. Deep learning methodologies have emerged as effective tools for detecting malware, given their capability to analyze various characteristics of malicious programs that can disrupt system operations by altering data. Numerous researchers have employed convolutional neural networks for classifying data, extracting essential features, and isolating genetic sequences indicative of malicious applications, thereby facilitating network training. Moreover, deep learning techniques are instrumental in identifying biological attributes such as personal identification numbers (PINs) and passwords, recognizing user voices or images, and analyzing behavior-based licenses. At this stage, techniques derived from recurrent neural networks (RNNs), including gated recurrent units and long short-term memory, are often deployed.

Device security stands as a pivotal concern within the realm of cybersecurity, wherein heightened security measures necessitate increased interactions between humans and electronic environments. Deep learning techniques play a crucial role in safeguarding data, systems, and applications. Renowned for their exceptional performance in processing 2D and 3D media data as well as vast datasets, these techniques are

extensively employed in image and video processing. In the cybersecurity domain, deep learning endeavors to discern the suitability of received data for supervised or unsupervised techniques and to evaluate the influence of prior knowledge on subsequent insights. Moreover, deep learning assesses system performance in addressing problems across both one- and multi-dimensional examples. Scholars are actively exploring deep learning methodologies to devise solutions for numerous cybersecurity challenges [68–70].

Table 5: Scholarly works employing deep learning methodologies for the detection of attacks and malicious software.

Article	Purpose	Technique use	Most Suitable Effect
[71]	Protect autonomous vehicle systems from attacks and ensure control.	CNN and CNN-LSTM	Achieves a remarkably high accuracy exceeding 97% in identifying attack messages and preventing their display on vehicle screens.
[72]	Real-time intrusion detection in vehicular data encompassing cyber and physical processes.	LR, SVM, RF, DT, MLP, and RNN	RNN exhibits the best accuracy of 79.3% in detecting malware, denial-of-service (DoS) attacks, and command injections.
[73]	Predict software vulnerabilities and identify accessible features at an early stage.	ExBERT framework	ExBERT framework reveals over 46,000 vulnerabilities with a prediction accuracy surpassing 91%.
[74]	Analyze and detect attacks using URLs on edge devices, safeguarding	Multiple concurrent deep models	These models achieve a high accuracy of over 99% in detecting normal

	data in cloud-Internet of Things (IoT) systems.		requests.
[75]	Develop an intrusion detection application and protect computer systems.	MLP and PID	MLP and PID attain an accuracy of 98.96% in intrusion detection and understanding attack types.
[76]	Detect intrusions and analyze network anomalies.	K-NN and DNN	DNN achieves over 92% accuracy in intrusion detection.
[77]	Establish an intrusion detection approach for cyber-attack security and classification.	RNN	RNN achieves the highest accuracy of 98.27%.
[78]	Enhance simulation training methods to detect anomaly intrusion via the Internet.	RBM and DBM	RBM and DBM demonstrate an exceptional accuracy of 97.9% in detecting anomaly intrusions.
[79]	Identify malicious programs in multi-cloud healthcare systems using the MUSE model.	DHSNN	DHSNN achieves excellent training and testing accuracies ranging from 95% to 100% in detecting new attacks on dataflows.
[80]	Classify a traffic detection system and network fault identification or intrusion detection system.	CNN	CNN reports an efficacy of over 99%.

3. Conclusion and Future Work

Artificial intelligence stands as a cornerstone of the Fourth Industrial Revolution, poised to continue its profound impact on society due to its manifold benefits. However, to strike a harmonious balance between technological advancement and fundamental human values, the nexus between artificial intelligence and cybersecurity requires thorough exploration. It is imperative to harness modern technologies within virtual environments and social networking platforms to safeguard user privacy and information. The evolution of artificial intelligence capabilities must parallel the emergence of novel applications, fostering digital collaboration among nations and leveraging the integration of digital technologies into physical settings. Facilitating unrestricted access to data for researchers, while upholding user privacy, is essential for training artificial intelligence algorithms and conducting data analysis on a broader scale. Increased financial and ethical investments in machine learning and deep learning are vital to fortify the privacy of social media users and mitigate data breaches. Continuous training for cybersecurity professionals on cutting-edge technologies, hacker tactics, and malware behavior is imperative. Furthermore, stringent penalties and fines should be enforced for the misuse of artificial intelligence techniques and unauthorized privacy breaches. Future research endeavors should focus on implementing these techniques for predictive and classification purposes in cybersecurity, thus optimizing their efficacy.

Conflict of Interest

The authors declare no conflict of interest.

References

- [1] N. Bhalaji, "Reliable Data Transmission with Heightened Confidentiality and Integrity in IOT Empowered Mobile Networks," *Journal of IoT in Social, Mobile, Analytics, and Cloud*, vol. 2, no. 2, 106–117, 2020, doi:10.36548/jismac.2020.2.004.
- [2] J. Budd, B.S. Miller, E.M. Manning, V. Lampos, M.Z. et al., "Digital technologies in the public-health response to COVID-19," *Nature Medicine*, vol. 26, 1183–1192, 2020, doi:10.1038/s41591-020-1011-4.
- [3] K. Leung, J.T. Wu, G.M. Leung, "Real-time tracking and prediction of COVID-19 infection using digital proxies of population mobility and mixing," *Nature Communications*, vol. 12, no. 1501, 1–8, 2021, doi:10.1038/s41467-021-21776-2.
- [4] S. Shrestha, S. Haque, S. Dawadi, R.A. Giri, "Preparations for and practices of online education during the Covid-19 pandemic: A study of Bangladesh and Nepal," *Education and Information Technologies*, vol. 27, 243–265, 2021, doi:10.1007/s10639-021-10659-0.
- [5] M. Ssenyonga, "Imperatives for post COVID-19 recovery of Indonesia's education, labor, and SME sectors," *Cogent Economics & Finance*, vol. 9, no. 1, 1–51, 2021, doi:10.1080/23322039.2021.1911439.
- [6] H. Saleous, M. Ismail, S.H. AlDaajeh, N. Madathil, S. Alrabae, "COVID-19 pandemic and the cyberthreat landscape: Research challenges and opportunities," *Digital Communications and Networks*, vol. In press, , 2022, doi:10.1016/j.dcan.2022.06.005.
- [7] H.S. Lallie, L.A. Shepherd, J.R.C. Nurse, A. Erola, G.E. et al., "Cyber security in the age of COVID-19: A timeline and analysis of cyber-crime and cyber-attacks during the pandemic," *Computers & Security*, vol. 105, 102248, 2021, doi:10.1016/j.cose.2021.102248.
- [8] J. Li, "Cyber security meets artificial intelligence: a survey," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, 1462–1474, 2019, doi:10.1631/FITEE.1800573.
- [9] Z. Zhang, H. Ning, F. Shi, F. Farha, Y. Xu, F.Z. et al., "Artificial intelligence in cyber security: research advances, challenges, and opportunities," *Artificial Intelligence Review*, vol. 55, 1029–1053, 2021, doi:10.1007/s10462-021-09976-0.
- [10] M.M. Mijwil, "Implementation of Machine Learning Techniques for the Classification of Lung X-Ray Images Used to Detect COVID-19 in Humans," *Iraqi Journal of Science*, vol. 62, no. 6, 2099–2109, 2021, doi:10.24996/ijis.2021.62.6.35.
- [11] J. Cáceres-Hidalgo, D. Avila-Pesantez, "Cybersecurity Study in 5G Network Slicing Technology: A Systematic Mapping Review," in *Proceedings of IEEE Fifth Ecuador Technical Chapters Meeting*, IEEE, Cuenca, Ecuador: 1–6, 2021, doi:10.1109/ETCM53643.2021.9590742.
- [12] T. Ghosh, H. Al Banna, S. Rahman, S. Kaiser, M.M. et al., "Artificial intelligence and internet of things in screening and management of autism spectrum disorder," *Sustainable Cities and Society*, vol. 74, 103189, 2021, doi:10.1016/j.scs.2021.103189.
- [13] A. Adadi, M. Lahmer, S. Nasiri, "Artificial Intelligence and COVID-19: A Systematic umbrella review and roads ahead," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, 5898–5920, 2022, doi:10.1016/j.jksuci.2021.07.010.
- [14] M. Abdullahi, Y. Baashar, H. Alhussian, A. Alwadain, N.A. et al., "Detecting Cybersecurity Attacks in Internet of Things Using Artificial Intelligence Methods: A Systematic Literature Review," *Electronics*, vol. 11, no. 2, 1–27, 2022, doi:10.3390/electronics11020198.
- [15] I.F. Kilincer, F. Ertam, A. Sengur, "Machine learning methods for cyber security intrusion detection: Datasets and comparative study," *Computer Networks*, vol. 188, 107840, 2021, doi:10.1016/j.comnet.2021.107840.
- [16] S. Kuipers, M. Schonheit, "Data Breaches and Effective Crisis Communication: A Comparative Analysis of Corporate Reputational Crises," *Corporate Reputation Review*, vol. 25, 176–197, 2021, doi:10.1057/s41299-021-00121-9.
- [17] N. Rawindaran, A. Jayal, E. Prakash, C. Hewage, "Cost Benefits of Using Machine Learning Features in NIDS for Cyber Security in UK Small Medium Enterprises (SME)," *Future Internet*, vol. 13, no. 8, 1–36, 2021, doi:10.3390/fi13080186.
- [18] F. Quayyum, D.S. Cruzes, L. Jaccheri, "Cybersecurity awareness for children: A systematic literature review," *International Journal of Child-Computer Interaction*, vol. 30, 100343, 2021, doi:10.1016/j.ijcci.2021.100343.
- [19] P. Formosa, M. Wilson, D. Richards, "A principlist framework for cybersecurity ethics," *Computers & Security*, vol. 109, 102382, 2021, doi:10.1016/j.cose.2021.102382.
- [20] I.H. Sarker, H. Furhad, R. Nowrozy, "AI-Driven Cybersecurity: An Overview, Security Intelligence Modeling and Research Directions," *SN Computer Science*, vol. 2, no. 173, 2021, doi:10.1007/s42979-021-00557-0.
- [21] E. Fosch-Villaronga, T. Mahler, "Cybersecurity, safety and robots: Strengthening the link between cybersecurity and safety in the context of care robots," *Computer Law & Security Review*, vol. 41, 105528, 2021, doi:10.1016/j.clsr.2021.105528.

- [22] P. Sharma, S. Jain, S. Gupta, V. Chamola, "Role of machine learning and deep learning in securing 5G-driven industrial IoT applications," *Ad Hoc Networks*, vol. 123, 102685, 2021, doi:10.1016/j.adhoc.2021.102685.
- [23] A. Rehman, T. Saba, T. Mahmood, Z. Mehmood, M.S. et al., "Data hiding technique in steganography for information security using number theory," *Journal of Information Science*, vol. 45, no. 6, 767–778, 2018, doi:10.1177/0165551518816303.
- [24] G. Hale, C. Bartlett, "Managing the Regulatory Tangle: Critical Infrastructure Security and Distributed Governance in Alberta's Major Traded Sectors," *Journal of Borderlands Studies*, vol. 34, no. 2, 257–279, 2018, doi:10.1080/08865655.2017.1367710.
- [25] Y. Wang, A. Smahi, H. Zhang, H. Li, "Towards Double Defense Network Security Based on Multi-Identifier Network Architecture," *Sensors*, vol. 22, no. 3, 1–17, 2022, doi:10.3390/s22030747.
- [26] D.G. Broo, U. Boman, M. Törngren, "Cyber-physical systems research and education in 2030: Scenarios and strategies," *Journal of Industrial Information Integration*, vol. 21, 100192, 2021, doi:10.1016/j.jii.2020.100192.
- [27] M.M. Mijwil, "Malware Detection in Android OS Using Machine Learning Techniques," *Data Science and Applications*, vol. 3, no. 2, 5–9, 2020.
- [28] U. Urooj, B.A.S. Al-rimy, A. Zainal, F.A. Ghaleb, M.A. Rassam, "Ransomware Detection Using the Dynamic Analysis and Machine Learning: A Survey and Research Directions," *Applied Sciences*, vol. 12, no. 1, 1–45, 2021, doi:10.3390/app12010172.
- [29] A.F. AL-Otaibi, E.S. Alsuwat, "A Study on Social Engineering Attacks: Phishing Attack," *International Journal of Recent Advances in Multidisciplinary Research*, vol. 7, no. 11, 6374–6379, 2020.
- [30] A. Narote, V. Zutshi, A. Potdar, R. Vichare, "Detection of DDoS Attacks using Concepts of Machine Learning," *International Journal for Research in Applied Science & Engineering Technology*, vol. 10, no. VI, 390–403, 2022.
- [31] N. Bedeković, L. Havaš, T. Horvat, D. Crčić, "The Importance of Developing Preventive Techniques for SQL Injection Attacks," *Tehnički Glasnik*, vol. 16, no. 4, 523–529, 2022, doi:10.31803/tg-20211203090618.
- [32] U.K. Singh, C. Joshi, D. Kanellopoulos, "A framework for zero-day vulnerabilities detection and prioritization," *Journal of Information Security and Applications*, vol. 46, 164–172, 2019, doi:10.1016/j.jisa.2019.03.011.
- [33] Y. Wang, A. Zhou, S. Liao, R. Zheng, R. Hu, L. Zhang, "A comprehensive survey on DNS tunnel detection," *Computer Networks*, vol. 179, 108322, 2021, doi:10.1016/j.comnet.2021.108322.
- [34] Y. Zhou, P. Wang, "An ensemble learning approach for XSS attack detection with domain knowledge and threat intelligence," *Computers & Security*, vol. 82, 261–269, 2019, doi:10.1016/j.cose.2018.12.016.
- [35] J. He, C. Chang, P. He, M.S. Pathan, "Network Forensics Method Based on Evidence Graph and Vulnerability Reasoning," *Future Internet*, vol. 8, no. 4, 1–18, 2016, doi:10.3390/fi8040054.
- [36] M.P. Singh, A. Bhandari, "New-flow based DDoS attacks in SDN: Taxonomy, rationales, and research challenges," *Computer Communications*, vol. 15, 509–527, 2020, doi:10.1016/j.comcom.2020.02.085.
- [37] J.L.G. Torres, C.A. Catania, E. Veas, "Active learning approach to label network traffic datasets," *Journal of Information Security and Applications*, vol. 49, 102388, 2019, doi:10.1016/j.jisa.2019.102388.
- [38] S. Choudhary, N. Kesswani, "Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT," *Procedia Computer Science*, vol. 167, 1561–1573, 2020, doi:10.1016/j.procs.2020.03.367.
- [39] L. Dhanabal, S.P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 6, 446–452, 2015.
- [40] B. Bouyeddou, F. Harrou, B. Kadri, Y. Sun, "Detecting network cyber-attacks using an integrated statistical approach," *Cluster Computing*, vol. 24, 1435–1453, 2020, doi:10.1007/s10586-020-03203-1.
- [41] M. Idhammad, K. Afdel, M. Belouch, "Semi-supervised machine learning approach for DDoS detection," *Applied Intelligence*, vol. 48, 3193–3208, 2018, doi:10.1007/s10489-018-1141-2.
- [42] N. Koroniotis, N. Moustafa, E. Sitnikova, B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, 779–796, 2019, doi:10.1016/j.future.2019.05.041.
- [43] I.H. Sarker, "Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective," *SN Computer Science*, vol. 2, no. 154, 1–16, 2021, doi:10.1007/s42979-021-00535-6.
- [44] S.M. Kasongo, Y. Sun, "Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset," *Journal of Big Data*, vol. 7, no. 105, 1–20, 2020, doi:10.1186/s40537-020-00379-6.
- [45] R.T. S., R. Sathya, "Ensemble Machine Learning Techniques for Attack Prediction in NIDS Environment," *Iraqi Journal For Computer Science and Mathematics*, vol. 3, no. 2, 78–82, 2022, doi:10.52866/jjcs.2022.02.01.008.
- [46] Y. Niu, A. Korneev, "Identification Method of Power Internet Attack Information Based on Machine Learning," *Iraqi Journal For Computer Science and Mathematics*, vol. 3, no. 2, 1–7, 2022, doi:10.52866/jjcs.2022.02.01.001.
- [47] M.M. Mijwil, E.A. Al-Zubaidi, "Medical Image Classification for Coronavirus Disease (COVID-19) Using Convolutional Neural Networks," *Iraqi Journal of Science*, vol. 62, no. 8, 2740–2747, 2021, doi:10.24996/ij.2021.62.8.27.
- [48] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, M. Portmann, "Feature extraction for machine learning-based intrusion detection in IoT networks," *Digital Communications and Networks*, vol. In press, , 2022, doi:10.1016/j.dcan.2022.08.012.
- [49] M.A. Teixeira, T. Salman, M. Zolanvari, R. Jain, N. Meskin, M. Samaka, "SCADA System Testbed for Cybersecurity Research Using Machine Learning Approach," *Future Internet*, vol. 10, no. 8, 1–15, 2018, doi:10.3390/fi10080076.
- [50] K. Aggarwal, M.M. Mijwil, Sonia, A.H. Al-Mistarehi, S. Alomari, M. Gök, A.M. Alaabdin, S.H. Abdulrhman, "Has the Future Started? The Current Growth of Artificial Intelligence, Machine Learning, and Deep Learning," *Iraqi Journal for Computer Science and Mathematics*, vol. 3, no. 1, 115–123, 2022, doi:10.52866/jjcs.2022.01.01.013.
- [51] L.F. Maimó, A.H. Celdrán, A.L.P. Gómez, F.J.G. Clemente, J. Weimer, I. Lee, "Intelligent and Dynamic Ransomware Spread Detection and Mitigation in Integrated Clinical Environments," *Sensors*, vol. 19, no. 5, 1–31, 2019, doi:10.3390/s19051114.
- [52] V.M. Rios, P.R.M. Inácio, D. Magoni, M.M. Freire, "Detection of reduction-of-quality DDoS attacks using Fuzzy Logic and machine learning algorithms," *Computer Networks*, vol. 186, 107792, 2021, doi:10.1016/j.comnet.2020.107792.
- [53] Y. Li, J. Xia, S. Zhang, J. Yan, X. Ai, K. Dai, "An efficient intrusion detection system based on support vector machines and gradually feature removal method," *Expert Systems with Applications*, vol. 39, no. 1, 424–430, 2012, doi:10.1016/j.eswa.2011.07.032.
- [54] W. Meng, W. Li, L. Kwok, "Design of intelligent KNN-based

- alarm filter using knowledge-based alert verification in intrusion detection,” *Security and Communication Networks*, vol. 8, no. 18, 3883–3895, 2015, doi:10.1002/sec.1307.
- [55] A. Mahindru, A.L. Sangal, “MLDroid—framework for Android malware detection using machine learning techniques,” *Neural Computing and Applications*, vol. 33, 5183–5240, 2020, doi:10.1007/s00521-020-05309-4.
- [56] H. Zuhair, A. Selamat, “RANDS: A Machine Learning-Based Anti-Ransomware Tool for Windows Platforms,” in *Advancing Technology Industrialization Through Intelligent Software Methodologies, Tools and Techniques*, 573–587, 2019, doi:10.3233/FAIA190081.
- [57] U. Adamu, I. Awan, “Ransomware Prediction Using Supervised Learning Algorithms,” in *Proceedings of International Conference on Future Internet of Things and Cloud*, Istanbul, Turkey: 1–6, 2019, doi:10.1109/FiCloud.2019.00016.
- [58] S. Puthran, K. Shah, “Intrusion Detection Using Improved Decision Tree Algorithm with Binary and Quad Split,” in *Proceedings of International Symposium on Security in Computing and Communication*, 427–438, 2016, doi:10.1007/978-981-10-2738-3_37.
- [59] J. Zhang, M. Zulkernine, A. Haque, “Random-Forests-Based Network Intrusion Detection Systems,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 38, no. 5, 649–659, 2008, doi:10.1109/TSMCC.2008.923876.
- [60] F. Musumeci, A.C. Fidanci, F. Paolucci, F. Cugini, M. Tornatore, “Machine-Learning-Enabled DDoS Attacks Detection in P4 Programmable Networks,” *Journal of Network and Systems Management*, vol. 30, no. 21, 2021, doi:10.1007/s10922-021-09633-5.
- [61] A.M. Chandrasekhar, K. Raghuveer, “Confederation of FCM clustering, ANN and SVM techniques to implement hybrid NIDS using corrected KDD cup 99 dataset,” in *Proceedings of International Conference on Communication and Signal Processing*, Melmaruvathur, India: 1–6, 2014, doi:10.1109/ICCSP.2014.6949927.
- [62] S. Ahmed, Z.A. Abbood, H.M. Farhan, B.T. Yaseen, M.R. Ahmed, A.D. Duru, “Speaker Identification Model Based on Deep Neural Networks,” *Iraqi Journal For Computer Science and Mathematics*, vol. 3, no. 1, 108–114, 2022, doi:10.52866/ijcsm.2022.01.01.012.
- [63] A.K. Faieq, M.M. Mijwil, “Prediction of Heart Diseases Utilising Support Vector Machine and Artificial Neural Network,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 1, 374–380, 2022, doi:10.11591/ijeecs.v26.i1.pp374-380.
- [64] M.M. Mijwil, R.A. Abttan, A. Alkhazraji, “Artificial intelligence for COVID-19: A Short Article,” *Asian Journal of Pharmacy, Nursing and Medical Sciences*, vol. 10, no. 1, 1–6, 2022, doi:10.24203/ajpnms.v10i1.6961.
- [65] K. Shaukat, S. Luo, V. Varadharajan, I.A. Hameed, S. Chen, et al., “Performance Comparison and Current Challenges of Using Machine Learning Techniques in Cybersecurity,” *Energies*, vol. 13, no. 10, 1–27, 2020, doi:10.3390/en13102509.
- [66] D. Chen, P. Wawrzynski, Z. Lv, “Cyber security in smart cities: A review of deep learning-based applications and case studies,” *Sustainable Cities and Society*, vol. 66, 102655, 2021, doi:10.1016/j.scs.2020.102655.
- [67] P. Suresh, K. Logeswaran, R.M. Devi, K. Sentamilselvan, G.K. Kamalam, H. Muthukrishnan, Contemporary survey on effectiveness of machine and deep learning techniques for cyber security, 177–200, 2022, doi:10.1016/B978-0-323-85209-8.00007-9.
- [68] M. Taseer, H. Ghafory, “SQL Injection Attack Detection Using Machine Learning Algorithm,” *Mesopotamian Journal of CyberSecurity*, 5–17, 2022, doi:10.58496/MJCS/2022/002.
- [69] I.E. Salem, M. Mijwil, A.W. Abdulqader, M.M. Ismaeel, A. Alkhazraji, A.M.Z. Alaabdin, “Introduction to The Data Mining Techniques in Cybersecurity,” *Mesopotamian Journal of CyberSecurity*, 28–37, 2022, doi:10.58496/MJCS/2022/004.
- [70] R.T. Rasheed, Y. Niu, S.N. Abd, “Harmony Search for Security Enhancement,” *Mesopotamian Journal of CyberSecurity*, 5–8, 2021, doi:10.58496/MJCS/2021/002.
- [71] T.H.H. Aldhyani, H. Alkahtani, “Attacks to Automotous Vehicles: A Deep Learning Algorithm for Cybersecurity,” *Sensors*, vol. 22, no. 1, 1–20, 2022, doi:10.3390/s22010360.
- [72] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon, et al., “Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning,” *IEEE Access*, vol. 6, 3491–3508, 2017, doi:10.1109/ACCESS.2017.2782159.
- [73] J. Yin, M. Tang, J. Cao, H. Wang, “Apply transfer learning to cybersecurity: Predicting exploitability of vulnerabilities by description,” *Knowledge-Based Systems*, vol. 210, 106529, 2020, doi:10.1016/j.knsys.2020.106529.
- [74] Z. Tian, C. Luo, J. Qiu, X. Du, M. Guizani, “A Distributed Deep Learning System for Web Attack Detection on Edge Devices,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, 1963–1971, 2020, doi:10.1109/TII.2019.2938778.
- [75] A. Thirumalairaj, M. Jeyakarthis, “Perimeter Intrusion Detection with Multi Layer Perception using Quantum Classifier,” in *Proceedings of International Conference on Inventive Systems and Control*, Coimbatore, India: 1–6, 2020, doi:10.1109/ICISC47916.2020.9171159.
- [76] K. Atefi, H. Hashim, M. Kassim, “Anomaly Analysis for the Classification Purpose of Intrusion Detection System with K-Nearest Neighbors and Deep Neural Network,” in *Proceedings of Conference on Systems, Process and Control*, Melaka, Malaysia: 1–6, 2019, doi:10.1109/ICSPC47137.2019.9068081.
- [77] M. Almiani, A. AbuGhazleh, A. Al-Rahayfeh, S. Atiewi, A. Razaque, “Deep recurrent neural network for IoT intrusion detection system,” *Simulation Modelling Practice and Theory*, vol. 101, 102031, 2020, doi:10.1016/j.simpat.2019.102031.
- [78] K. Alrawashdeh, C. Purdy, “Toward an Online Anomaly Intrusion Detection System Based on Deep Learning,” in *Proceedings of International Conference on Machine Learning and Applications*, Anaheim, CA, USA: 1–6, 2016, doi:10.1109/ICMLA.2016.0040.
- [79] L. Gupta, T. Salman, A. Ghubaish, D. Unal, A.K. Al-Ali, R. Jain, “Cybersecurity of multi-cloud healthcare systems: A hierarchical deep learning approach,” *Applied Soft Computing*, vol. 118, 108439, 2022, doi:10.1016/j.asoc.2022.108439.
- [80] W. Wang, M. Zhu, X. Zeng, X. Ye, Y. Sheng, “Malware traffic classification using convolutional neural network for representation learning,” in *Proceedings of International Conference on Information Networking*, Da Nang, Vietnam: 1–6, 2017, doi:10.1109/ICOIN.2017.7899588.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



Engr. Ramsha Khalid has done her bachelor's degree from Lahore College for Women University, Lahore in 2018. She has done her master's degree from University of Lahore, Lahore in 2022. She is working as Lecturer at University of Sialkot since 2019. Her area of interest includes Computer & Communication Networks, Machine Learning, Artificial Intelligence,

Cyber Security, Control Systems and Renewable Energy Systems. Recently she has published a conference paper in IEEE INMIC'23 held in University of Central Punjab, Lahore as a first author.



Engr. M. Naqi Raza has done his bachelor's degree from University of Gujrat, Gujrat in 2018. He has done his master's degree from University of Sialkot, Sialkot in 2024.

He is working as Junior Lecturer at University of Sialkot since 2019. His area of interest includes Power Generation (Conventional and Renewable), Wind Power Generation and Utilization, Optimization of Wind Energy, Solar Energy, Solar Power Applications, Electric Vehicles (PHEVs).

Comprehensive Analysis of Software-Defined Networking: Evaluating Performance Across Diverse Topologies and Investigating Topology Discovery Protocols

Nikolaos V. Oikonomou^{*1}, Dimitrios V. Oikonomou², Eleftherios Stergiou¹, Dimitrios Liarokapis¹

¹Department of Informatics & Telecommunications, University of Ioannina, Arta, 47150, Greece

²Department of Regional & Cross Border Studies, University of Western Macedonia, Kozani, 50100, Greece

*Corresponding author: Nikolaos V. Oikonomou, University of Ioannina Department of Informatics & Telecommunications, haikos13@gmail.com

ABSTRACT: Software-defined networking (SDN) represents an innovative approach to network architecture that enhances control, simplifies complexity, and improves operational efficiencies. This study evaluates the performance metrics of SDN frameworks using the Mininet simulator on virtual machines hosted on a Windows platform. The research objectives include assessing system performance across various predefined network topologies, investigating the impact of switch quantities on network performance, measuring CPU consumption, evaluating RAM demands under different network loads, and analyzing latency in packet transmission. Methods involved creating and testing different network topologies, including basic, hybrid, and custom, with the Mininet simulator. Performance metrics such as CPU and RAM usage, latency, and bandwidth were measured and analyzed. The study also examined the performance and extendibility of the OpenFlow Data Path (OFDP) protocol using the POX controller. Results indicate that balanced tree topologies consume the most CPU and RAM, while linear topologies are more efficient. Random topologies offer adaptability but face connection reliability issues. The POX controller and OFDP protocol effectively manage SDN network scalability. This research aims to analyze performance in a manner consistent with numerous previous studies, underscoring the importance of performance metrics and the scale of the network in determining the efficiency and reliability of SDN implementations. By benchmarking various topologies and protocols, the research offers a valuable reference for both academia and industry, promoting the development of more efficient SDN solutions. Understanding these performance metrics helps network administrators make informed decisions about implementing SDN frameworks to improve network performance and reliability.

Keywords: Network Architecture, Efficiency, SDN Controllers, Network Simulation, OpenFlow Protocol

1. Introduction

Networks are all around us, integral to our lives and daily routines. Most of the needs of a modern, technologically advanced society require strong and reliable networks. The demand for efficient networks is increasing exponentially with the passage of years and technological development. Nowadays, most people from all age groups use networks daily, and their quality of life depends on these networks, even if this is not immediately apparent. The COVID-19 pandemic that the world experienced from the beginning of 2020 changed many aspects of network usage. People were forced to spend

much more time at home. This situation led people to find smart ways to meet most of their daily needs within the walls of their homes. Thus, the concept of the network in general came to everyone's doorstep in the form of the largest known global network, the internet. Young people had to be educated remotely using the internet. Adults were mostly required to work from a distance, and the elderly and vulnerable groups had to seek their care and support in a different way with the help of the internet. This situation led to an unprecedented increase not only in the number of internet users but also in the number of different devices each user employs to access it. As a result, some weaknesses in the existing global networks were revealed, and new ones were created. Network providers

and researchers realized that the capacity, speed, management, and reliability of networks needed to be increased due to the excessive load. Traditional networks that had been used for several years began to collapse because they primarily relied on physical infrastructures. Technologies such as SDN, which had been around for the last 10 years (mainly from 2013 onwards), began to be more extensively researched and implemented in global networks to strengthen them and ensure the smooth survival of the world. SDN enhances the capabilities of the network and simplifies its structure, with their main goal being more efficient network management. In this study, the architecture and structure of SDN networks will be examined. Network topologies will be created through simulations, and their operation and performance will be analyzed. Specifically, the performance between different topologies will be compared. It will be shown how the total number of switches, which are a fundamental pillar in the architecture of SDN, affects and burdens the overall performance of the network. The main measurements to be taken to draw accurate conclusions are CPU usage, RAM memory, and the delay in packet transfer between nodes. To achieve the above in the form of simulation, the Mininet simulator will be used on a computer with a Windows operating system. Additional software will be used in conjunction with Mininet to ensure the integrity and number of results. In this way, the behavior and adaptability of SDN networks will be studied. The controller used in Mininet will be POX, and further analysis will be done on the topology creation protocol OFDP, through which virtual networks will be studied below and their performance and scalability examined. The aim is to draw conclusions about the operation of SDN with the POX controller, specifically the use of the OFDP protocol. The reason for using random topologies is their effects on traditional networks and raises the question of how these random graphs can affect OpenFlow as its evolution into an even more modern network topology creation protocol. Given that network technologies have a strong relationship with network graphs and consequently with the concept of graph theory in mathematics, an analytical model for computation, comparison, and prediction is established through simulations on realistic platforms so that faster implementation in real-time networks can be achieved. The remainder of the article is as follows: Related search, SDN controllers, SDN protocols, Software & Hardware specifications, experiment specifications, analysis of results and finally the total conclusions from this research [1].

2. Related Search

Several research publications have been made on the aspect of SDNs, using the POX controller as well as OFDP for creating topologies. However, few utilize random topologies for interpreting SDN performance. In our previous research, we studied the performance results of

Software-Defined Networking (SDN) tests conducted on standard network topologies using simulation. Concurrently, the performance of the standard topologies was compared with that of the random ones. Specifically, the performance measurements examined included: the setup and teardown time of the topology, the CPU and RAM usage of the system, and the delay in packet transfer between nodes. The entire study was conducted on a Windows computer using a virtual machine to run a Mininet simulator, similar to what we will use in the present work. From the meticulous analysis of the results, the following are worth mentioning: (i) the total number of switches in an SDN architecture has a significant impact on CPU load. (ii) RAM usage depends on the number of host computers and in cases of excessive load, it shows a much greater increase compared to CPU usage. (iii) The overall performance significantly depends on the type of topology and its properties. The experiments then were conducted on a typical and limited range of devices [2].

In his work, Guo created various types of network topologies for analysis, including ring topology, tree topology, and random Erdos-Renyi model topologies. In the randomly created networks, the probability of an edge between any two vertices was set at 0.4. Thus, these random networks were mostly dense networks with a short average path length. The ring networks had the longest average path length among the three topologies, while the tree topology was intermediate; all experiments were repeated 100 times. All nodes were subject to a common failure probability. Fifty nodes were used to create the three types of networks. The results show that the expected resilience of the network is inversely proportional to the average path length of the network topology, hence random topology networks perform better, and ring networks are less resilient [3].

In another study, the performance of the proposed discovery mechanism, which primarily relies on the OFDP protocol regarding the overall load, was analyzed. Various topologies were examined, focusing on random networks based on the Erdős-Rényi model. The study highlighted that researchers' efforts are concentrated on reducing the number of messages reaching the controller. However, the performance and scalability of SDN networks depend more on other factors, such as CPU load, memory usage, network topology, and the time required for topology discovery. The scalability of OFDP and OFDPv2 for a wide range of random networks based on the Erdős-Rényi model was tested. Experimental results showed that the protocols consume almost equal resources (CPU and RAM), while OFDP requires more

time for topology discovery than OFDPv2 for the same topology [4].

3. SDN Controllers

3.1. General SDN Information

Software-Defined Networking (SDN) is a networking approach that uses software-based controllers or APIs to communicate with the underlying hardware infrastructure and direct traffic within a network. This model differs from traditional networks, which solely utilize hardware devices (routers and switches) to manage network traffic. SDN can create and manage a virtual network or control a traditional network through software. While network virtualization allows the segmentation of different virtual networks on a single physical network and the connection of devices across various physical networks to form a single virtual network, SDN enables a new method of controlling data packet routing through a central server. Consequently, SDN achieves the successful and functional separation of the Control Plane from the Forwarding Plane in a network. Below in Figure 1 the SDN architecture is depicted.

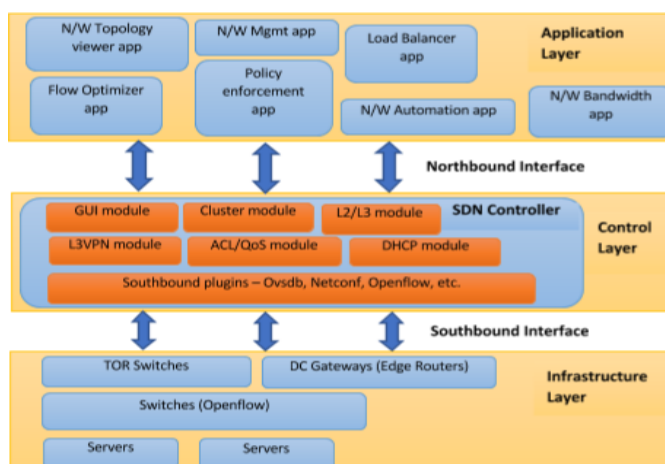


Figure 1: Schematic representation of the SDN architecture. Adapted from [5]

3.2. SDN Controller

A software-defined networking controller is a central element of the SDN architecture. It provides control over network elements in the managed domain. In networking, there are management, control, and data planes. An SDN controller offers management and control functions for network elements within the managed domain. This means that an SDN controller, based on network information and a set of predefined rules and policies, manages network elements and configures (or "programs") the data plane (i.e., directs data flow through the network). One of the key advantages of using an SDN controller is that it allows for more efficient network management, and changes to the network configuration can be applied from a central location instead of needing

to manually configure each individual network element. Additionally, an SDN controller can automate certain tasks, such as traffic management and security, which can reduce the risk of human error and improve the overall reliability of the network. SDN controllers provide an API known as the northbound interface, through which external applications or systems such as orchestration platforms can interact with the network. In such cases, an SDN controller translates application-level requirements (e.g., high-level network configuration description) into configurations specific to the supported network elements. SDN controllers can manage both physical network devices and software elements that perform network functions [6].

In summary, the main functions of an SDN controller include:

- Managing data flow within the managed network
- Providing an API for applications and other components (e.g., orchestration platforms) to interact with the network.
- Providing visibility into the network, enabling network performance monitoring and troubleshooting
- Automating network management tasks, such as provisioning new network elements and reconfiguring network paths

More specifically, the controller provides the following capabilities:

Southbound Support: Defined as how a controller interacts with network devices to achieve optimized traffic flow. There are various southbound protocols that can be used, each with specific functionalities such as field matching, network discovery with different protocols, etc. When supporting the southbound interface, implementers must consider not only the characteristics of the protocol but also potential extensions, newer versions, etc.

Northbound Support: Northbound APIs are used for network integration and programming and can be utilized by orchestration systems that cater to customers and third-party applications. It is crucial to ensure that a controller is properly developed for orchestrating communications between layers. For example, the controller should support orchestration systems for applications such as cloud services, not only for open-source controllers and protocols but also those provided by various vendors. These applications could also include traffic engineering or applications that collect data used for network management tasks. As we can see below in Figure 2 the differences between the structure of

Centralized and Distributed control path are presented [7].

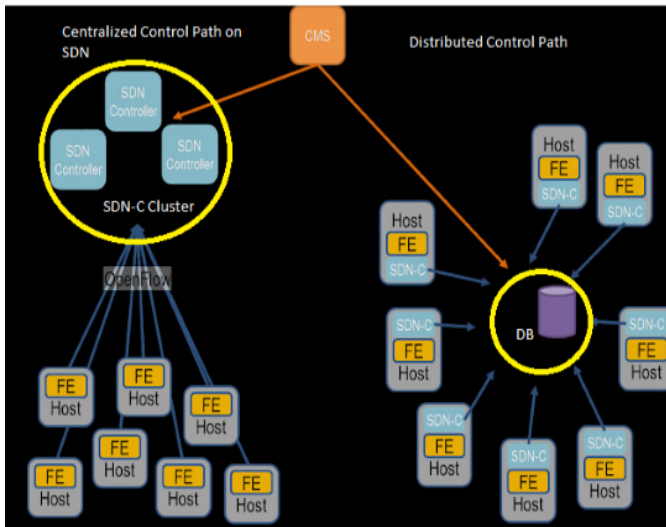


Figure 2: Comparison of Centralized and Distributed Architecture.

3.3. NOX/POX Controller

NOX was the first SDN controller. Initially developed by Nicira Networks, it was the first to support the OpenFlow protocol. Released to the research community in 2009, it laid the foundation for many SDN research projects. It was later expanded and supported at Stanford University with significant contributions from the University of California, Berkeley. Some popular NOX applications include SANE (an approach that represents the network as a file system) and Ethane. Today, NOX is considered inactive. Over the years, different versions of NOX have been introduced. These are known as NOX, NOX-MT, and POX. The new NOX only supports C++. It has a smaller application network compared to NOX but is much faster and has a much cleaner codebase. NOX-MT, introduced as a slightly modified version of the NOX controller, uses optimization techniques to introduce multi-threaded processing to improve the rate and response time of NOX. These optimization techniques include I/O batching to minimize general input/output overhead and others. POX is the latest version based on Python. The idea behind its development was to return NOX to its roots in C++ and develop a separate platform based on Python. It also features a Python OpenFlow interface, reusable element samples for path selection, topology discovery, etc. The primary goal of POX is research. Given that many research projects are by nature short-lived, the focus of POX developers is on good interfaces rather than API stability. In the current research, due to the multiple interfaces and the stability of the controller and because the Python language was used to create network topologies, POX was used [8].

Generally, the NOX controller provides a complete OpenFlow API using C++ and Python languages, uses asynchronous inputs/outputs (I/O), and is oriented towards operation on Linux, Ubuntu, and Debian systems. NOX is used both as a standalone controller and as a component-based framework for developing SDN applications. It is built on an event-based programming model and adopts a simple programming interface model that revolves around three pillars:

- Events
- Namespace
- Network view

Events can be generated either directly from OpenFlow messages or from NOX applications because of processing low-level events or other events generated by applications.

4. SDN Protocols

The SDN protocol is a set of standards and rules that define how SDN controllers and switches communicate with each other. Essentially, a protocol allows the SDN controller to configure the behavior of the switch, such as determining which packets should be forwarded to which ports and setting quality of service (QoS) parameters for different types of traffic. The most popular SDN protocol is OpenFlow.

4.1. OpenFlow Protocol

As mentioned, OpenFlow is the most widespread SDN protocol and defines the flow between the switch and the controller. It allows the controller to manage traffic forwarding between different network devices by controlling the switch's flow tables. This protocol was first developed by researchers at Stanford University in 2008 and was first adopted by Google in their backbone network in 2011-2012. It is now managed by the Open Networking Foundation (ONF). The latest version widely used in the industry is V1.5, while V2.0 is being refined. It is also often referred to as OFDP, meaning the OpenFlow Topology Discovery Protocol, because whether referred to as OpenFlow or OFDP, it automatically means the same function [9], [10].

OpenFlow is the standard southbound interface protocol used between the SDN controller and the switch. The SDN controller takes information from the applications and converts it into flow entries, which are fed into the switch via OpenFlow. It can also be used to monitor switch and port statistics in network management.

It is worth noting that the OpenFlow protocol is only installed between a controller and a switch. It does not affect the rest of the network. If a packet capture were to be taken between two switches in a network, both connected to the controller via another port, the packet capture would not reveal any OF messages between the switches. It is strictly for use between a switch and the controller. The rest of the network is not affected [11].

4.2. NetConf Protocol

NetConf is a protocol used in SDN for managing network devices such as routers and switches, providing a standardized way of configuring, monitoring, and managing these devices. It is an IETF standard and is based on XML data encoding and the SSH protocol for secure communication. The default TCP port assigned is 830. The NetConf server must listen for connections with the NetConf subsystem on this port.

With NetConf, network administrators can configure network devices programmatically using a standardized set of commands, rather than relying on proprietary interfaces for specific devices. This helps simplify network management and facilitates the automation of repetitive tasks such as deploying new network configurations or updates to hardware and software.

NetConf uses a client-server model, with the NetConf client sending requests to the device and the NetConf server responding with data or status updates. The protocol supports a range of functions, such as:

Retrieve: Retrieve specific data or configuration information from the device

Edit-config: Modify the device's configuration.

Commit: Apply changes to the device's configuration

Lock: Lock the device's configuration to prevent multiple managers from making conflicting changes

Unlock: Release the configuration lock

NetConf is often used in conjunction with YANG, a data modeling language that allows network administrators to describe network elements and their configurations in a structured and standardized way. Together, NetConf and YANG form a significant component of SDN, enabling greater automation and control in network management.

4.3. Open vSwitch Database Management Protocol (OVSDB)

OVSDB is a protocol used in SDN networks for managing Open vSwitch instances, which are software-

based switches that can be used in an SDN environment. OVSDB provides a standard way to configure and manage Open vSwitch instances, allowing network administrators to deploy and manage switches more automatically and programmatically. It defines a set of functions that can be used for querying and modifying the configuration of Open vSwitch instances, including creating, deleting, and modifying ports, interfaces, and VLANs.

OVSDB is based on a client-server model, with the OVSDB client sending requests to the OVSDB server, which responds with data or status updates. The protocol uses JSON data encoding and supports secure communication using TLS.

In an SDN environment, Open vSwitch instances can be used to forward traffic between different network devices and allow network administrators to manage traffic flows using a central SDN controller. OVSDB provides a standardized way to configure and manage these switches, facilitating the development and management of large-scale SDN networks.

4.4. Border Gateway Protocol (BGP)

BGP is a routing protocol commonly used in SDN networking environments to exchange routing information between different autonomous systems in large-scale networks. BGP is a path vector routing protocol that uses a network of interconnected autonomous systems to route traffic between different parts of the network.

In an SDN environment, BGP can be used to facilitate communication between different network elements, such as the SDN controller and network devices, or between different SDN controllers. BGP provides a standardized way to exchange routing information, allowing network managers to manage traffic flows and optimize network performance.

BGP uses a hierarchical routing table system, with each autonomous system maintaining its own routing table and exchanging updates with neighboring autonomous systems. The protocol supports both internal and external routing, allowing more efficient routing within a single autonomous system and between different autonomous systems [12].

4.5. Locator/Identifier Separation Protocol (LISP)

LISP is a protocol used in SDN to separate the network location of a device (essentially its IP address) from its identity or identifier. LISP provides a way to

assign multiple IP addresses to a device and allows routing of traffic based on the device's identity rather than its physical location.

In an SDN environment, LISP can be used to simplify network management and enable more efficient routing of traffic between different devices. By separating the device's identity from its location, LISP allows network administrators to move devices between different network locations without changing their IP addresses, simplifying network management and reducing the likelihood of errors [13].

4.6. Simple Network Management Protocol (SNMP)

SNMP is a protocol used in software-defined networking (SDN) for monitoring and managing network devices. SNMP provides a standardized way for network administrators to monitor the performance of network devices, such as switches and routers, and to configure them remotely.

In an SDN environment, SNMP can be used to monitor and manage network devices from a central SDN controller. SNMP allows network administrators to monitor a range of device metrics, such as CPU usage, memory usage, and network traffic, and to receive alerts when performance issues arise.

SNMP is based on a client-server model, with SNMP agents running on network devices and SNMP managers running on the SDN controller. The agents collect performance data and send it to the managers, who can then analyze the data and take actions to improve network performance.

SNMP is a widely used protocol in network management and is supported by a range of network device suppliers and SDN controllers. It can be used in conjunction with all the above-mentioned SDN protocols to enable more effective and flexible network management.

4.7. Link Layer Discovery Protocol (LLDP)

LLDP is a Layer 2, vendor-neutral protocol used for discovering and advertising network device information on a local area network (LAN). It allows network devices to exchange information about their identity, capabilities, and connections [14].

LLDP operates by sending and receiving LLDP frames, which are multicast packets transmitted on every network interface. LLDP frames contain TLV (Type-Length-Value) elements that carry specific information about the transmitting device, such as system name, port

description, system capabilities, and management addresses.

Key features and benefits of LLDP include:

Device Discovery: LLDP enables network devices to discover neighboring devices on the LAN, providing information about their identity, such as device type, vendor, and model.

Topology Discovery: By exchanging LLDP information, devices can gather details about the connections and topology of the network, including neighboring devices, port numbers, and connection speeds.

Automatic Configuration: LLDP can be used by network management systems to automatically configure network devices based on their discovered capabilities, simplifying network setup and reducing the efforts of manual configuration.

Troubleshooting and Monitoring: LLDP facilitates network troubleshooting by providing visibility into the network topology and device connectivity. It allows administrators to identify and locate devices, detect link failures, and monitor the status of connections.

LLDP is supported by a wide range of network devices, including switches, routers, wireless access points, and IP phones. It is often used in conjunction with other network protocols, such as SNMP, to enable comprehensive network management and monitoring.

It is important to note that LLDP is a Layer 2 protocol, and its functionality is limited to the local network segment. It does not route traffic nor provide visibility into the entire network.

4.8. Advantages of SDN

Software-defined networking (SDN) has emerged as a transformative approach to network architecture and management. By decoupling the control plane from the data plane and centralizing network control through software, SDN provides numerous benefits and impacts various industries. Key findings on SDN include:

- **Enhanced Network Flexibility:** SDN allows organizations to quickly provision, configure, and modify network services via software, leading to improved network flexibility. It enables dynamic allocation of network resources, making it easier to adapt to changing business needs and network traffic patterns [15].
- **Simplified Network Management:** SDN centralizes network management through a software-managed

controller, providing a single point of control and monitoring. This simplifies network management, reduces complexity, and enhances troubleshooting capabilities.

- **Scalability and Flexibility:** SDN offers scalability by abstracting network functionality from the underlying hardware. Organizations can more easily scale their networks by adding or reallocating resources according to needs. Furthermore, SDN allows flexibility in deploying new services and applications without significant changes to infrastructure.
- **Network Programmability:** SDN enables network programmability, allowing administrators to automate network functions and control network behavior through software. This programmability facilitates the development of innovative applications and services that can interact directly with the network.
- **Enhanced Security:** SDN provides enhanced security capabilities by leveraging centralized control and programmability. Security policies can be defined and enforced consistently across the network, making it easier to identify and respond to threats.
- **Cost Optimization:** SDN offers cost savings by reducing hardware dependencies and enhancing resource utilization. With the ability to dynamically control and distribute network resources, organizations can optimize their infrastructure, leading to better cost performance.
- **Innovation and Ecosystem Development:** SDN promotes innovation by enabling the development of new network services and applications. It encourages the development of an 'ecosystem' where vendors, developers, and researchers can collaborate to create new solutions and advance networking progress.
- **SD-WAN and Cloud Connectivity:** SDN plays a critical role in the adoption of software-defined wide area networks (SD-WAN) and in connecting on-premises networks to cloud environments. It simplifies the management of distributed networks, provides better visibility and control, and improves connectivity to cloud services.

4.8 challenges and issues

While SDN offers significant benefits, it also presents challenges, including interoperability among different SDN solutions, security concerns related to centralized control, the need for specialized personnel to manage and operate SDN environments, and the necessity for careful

planning, testing, and collaboration with experienced vendors to overcome these challenges.

SDN Protocols:

- SDN protocols play a critical role in the implementation and operation of software-defined networking (SDN) environments. These protocols define the communication and interaction between different elements of an SDN architecture, facilitating network control and management.
- OpenFlow is one of the most widely adopted SDN protocols. It provides a standard interface between the control layer and forwarding devices (switches). OpenFlow enables centralized network control by separating control logic from switches and allowing the controller to program forwarding rules. It has significantly contributed to the development and deployment of SDN solutions.

SDN Controllers:

- SDN controllers serve as the central intelligence of software-defined network (SDN) architectures. They are responsible for managing and orchestrating network resources, facilitating communication between the control layer and the data layer, and enabling network programmability.

Table 1: below presents the network protocols along with their pros and cons.

Table 1: Network protocols.

PROTOCOLS	PROS	CONS
OpenFlow	Fully customizable, scalable	Complex
NetConf	Simplicity, management	Limited Performance
OVSDB	Customizable, management	Few complex options
BGP	Usable across different networks, routing	Recommended only for very large networks
LISP	Simplicity, efficient traffic control	Limited capabilities
SNMP	Advanced control	Complex
LLDP	Wide range of device compatibility	Limited only to LAN networks

5. Software & Hardware specifications

In this section, we will analyze each tool used for this work. Specifically, both the hardware and software components will be discussed.

5.1. Hardware Specifications

Compared to previous related research where high-performance laptops, low-performance desktops, or even workstations were used, this research utilized a new high-performance desktop computer. This system offers the capability to implement larger virtual networks as well as optimized management and distribution of physical resources, allowing for improved performance and more efficient scaling of the networks that will be created. In the heart of the computing system used for this research, the Gigabyte B550M AORUS PRO motherboard with an AMD B550 chipset lays the foundation. This motherboard was chosen for its robust support for modern connectivity standards such as PCI EXPRESS 4.0, which is pivotal for high-performance setups required in advanced simulations and experiments. The AMD Ryzen 5 5600X processor, featuring a 7nm FinFET technology with 6 cores and 12 threads, is selected for its ability to handle extensive computations more effectively than comparable models used in preceding studies. Its overclocking ability up to 4.7 GHz facilitates faster processing of complex tasks, crucial for developing larger virtual networks and conducting intensive data analysis.

Additionally, the system is equipped with 32GB of DDR4 RAM at 3600 MHz in dual-channel configuration, providing ample bandwidth and speed necessary for managing multiple operations simultaneously, which is essential when testing the limits of network simulations and other resource-intensive applications. The AMD Radeon RX 6750 XT graphics card with 12GB of GDDR6 memory ensures smooth rendering of complex graphics and supports the visualization demands of the research, including the manipulation and analysis of high-dimensional data sets.

Storage is handled by a Kingston KC3000 NVMe SSD with a capacity of 2TB, leveraging PCI Express 4.0 technology to offer rapid data access speeds of up to 7000 MB/s, significantly reducing load times and improving the overall efficiency of data processing tasks. This storage solution is vital for handling large volumes of data generated during simulations, ensuring quick retrieval and processing that are imperative for maintaining workflow continuity during the research.

Together, these hardware specifications are meticulously chosen not only for their individual capabilities but also for their synergy, which ensures a high-performance, stable, and reliable computing environment capable of supporting the sophisticated software tools and simulations utilized in this research. In Table: 2 we have the technical specifications of our systems.

Table 2: Simulation system specifications.

Component	Specification
CPU	AMD Ryzen 5 5600X, 6 cores/12 threads, 4.7 GHz, 45W
RAM	32GB DDR4, 3600 MHz, Dual Channel
GPU	AMD Radeon RX 6750 XT, 12GB, PCI Express 4.0
Storage	Kingston KC3000, NVMe, PCI Express 4.0, 7GB/s

5.2. Software Specifications

In this section, the specifications of the system software used are analyzed. It is crucial not only to conduct research to use the correct software that can deliver the desired results but also to ensure that all software can work harmoniously together. Cohesion, relevance, and repeated checks on the outcomes that will be extracted are necessary. For the software setup in this research, specific tools have been meticulously selected to complement the powerful hardware configuration and to meet the specialized requirements of the study. The primary operating system used is Windows 11 Pro for Workstations, which offers essential features like the ReFS file system for enhanced data resilience and support for advanced hardware configurations, critical for maximizing the potential of the system's physical components.

Oracle's VirtualBox plays a key role by allowing the deployment of multiple operating systems on a single physical machine, which is crucial for testing different network configurations and software interactions in a controlled, isolated environment. This flexibility is vital for reproducing and manipulating network scenarios in the development of software-defined networking (SDN) solutions.

Additionally, Visual Studio Code is employed as the primary code editor due to its robust support for multiple programming languages and its integrated development environment (IDE) features like debugging, code completion, and Git integration. These features enhance the efficiency of writing and testing code, particularly Python scripts used for creating network topologies in the research.

Gephi, an open-source network visualization software, is used to analyze and visualize complex network structures, which helps in understanding the interactions within the network and identifying key patterns and anomalies. The ability to dynamically model network traffic and topology changes in real-time using Gephi significantly aids in the exploratory phase of the research.

Furthermore, the inclusion of specialized tools like PuTTY for secure remote session management, WinSCP for secure file transfer, and Xming for running X Window System applications on Windows, consolidates the software environment.

Together, these software tools form a cohesive ecosystem that supports the rigorous demands of the research, enabling sophisticated simulations, extensive data analysis, and effective management of resources across different stages of the project. Table 3 contains an analysis of all the software used.

Table 3: Simulation software presentation.

Software	Brief Description
Windows 11 Pro for Workstations	Operating system designed for high-tech hardware and workloads, with additional features for enhanced performance and reliability.
VirtualBox	Open-source virtualization software that allows running multiple operating systems on a single physical machine.
Mininet	Network emulator that facilitates the simulation and testing of Software-Defined Networks (SDN).
X-Ming	Free X-Window-System server for Windows that enables remote graphical user interfaces over a network.
WinSCP	Free and open-source SFTP, FTP, and SCP client for Windows that enables secure file transfers between local and remote computers.

PuTTY	Free terminal emulator, serial console, and network file transfer application for Windows that supports multiple network protocols.
Visual Studio Code	Free, open-source code editor developed by Microsoft, supporting a wide range of programming languages and tools.
Gephi	Open-source software for visualizing and exploring graphs and networks, ideal for analyzing complex networks.

6. Experiment Specifications

6.1. Network Topologies

The term topology defines the geometric representation of the connections in a network. We examined three categories of topologies.

- Basic
- Hybrid
- Custom

Specifically, for the basic topologies, the bus topology was selected, for the hybrid topologies, the balanced tree topology was chosen, and for the Custom, the random topology was used [16], [17].

6.1.1. Basic Topologies

There are many basic network topologies commonly used in computer networking. These include:

Bus topology: All devices are connected to a single communication line or cable, known as the bus. Data travels in both directions along the bus and all devices on the network can receive the same message simultaneously. Figure 3 depicts bus topology.

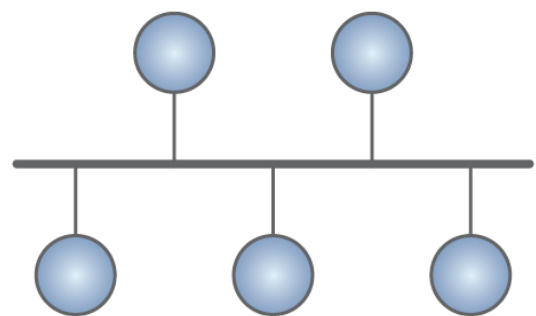


Figure 3: Example of bus topology.

Star topology: All devices are connected to a central hub or switch, and data flows through the hub or switch to reach its destination. Each device has an exclusive connection to the hub or switch, which can help reduce

network congestion and improve performance. Figure 4 depicts star topology.

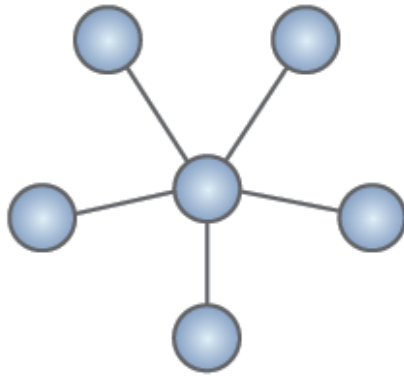


Figure 4: Example of star topology.

Ring topology: All devices are connected in a closed loop, with data flowing in one direction around the loop. Each device receives data from the previous device in the loop and sends data to the next device in the loop. The ring topology is depicted in Figure 5 below.

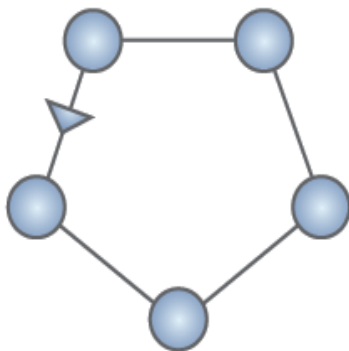


Figure 5: Example of ring topology.

Mesh topology: Each device is connected to every other device in the network, creating a fully interconnected network. This can provide high redundancy and fault tolerance but can be complex to manage and requires a lot of wiring. Figure 6 presents mesh topology.

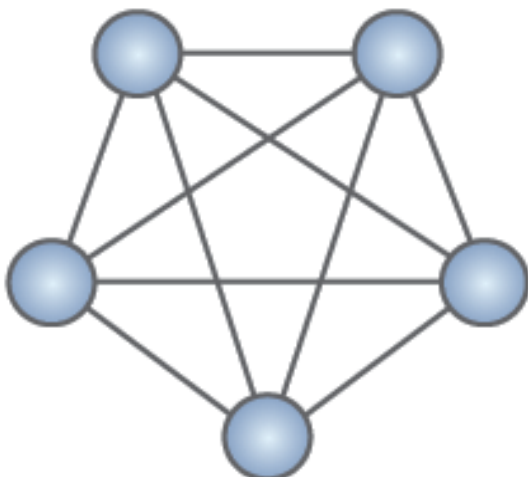


Figure 6: Example of mesh topology.

6.1.2. Hybrid Topologies

Hybrid Topology is a combination of two or more basic topologies, such as a star-bus topology or a ring-mesh topology. This can offer a balance between performance, redundancy, and ease of management.

Tree topology, also known as hierarchical topology, is a type of network topology based on a hierarchical structure. In this topology, multiple star topologies relate to a bus topology, creating a structure that resembles a tree. In a tree topology, the central bus acts as the main trunk of the tree, with multiple branches extending from it. Each branch is a separate star topology with a hub or switch at the center and multiple devices connected to it. This allows the creation of subnetworks within the larger network, with each subnet having its own exclusive hub or switch.

The main advantage of a tree topology is its scalability, as it can support many devices and subnetworks. It also provides a good balance between performance and redundancy, as each subnet can operate independently and problems in one subnet will not affect the rest of the network.

However, the main disadvantage of a tree topology is its complexity, as it requires a significant amount of cabling and configuration. It can also be difficult to troubleshoot and manage, as problems in one part of the network can affect the entire tree. Below in Figure 7 an example of hybrid topology can be found.

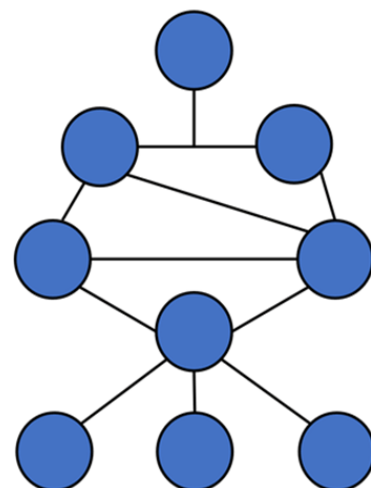


Figure 7: Example of hybrid tree topology.

A balanced tree topology is a specific type of tree topology where each branch of the tree has the same number of levels. This means that each subnet is of equal size and has the same number of devices connected to it. In a balanced tree topology, the central bus is connected to a set of level-1 switches, each of which is connected to

a set of level-2 switches, and so on, until the final level of switches is reached. Each switch in the tree has an equal number of branches connected to it, which helps balance the network traffic and avoid congestion.

6.1.3. Custom Topologies

Custom network topologies refer to network architectures designed to meet specific requirements or solve specific problems. They may be a combination of two or more basic topologies, or they may be entirely unique and tailored to a specific application or environment. Custom network topologies can be created by network designers and administrators using various networking devices and technologies, such as switches, routers, firewalls, load balancers, and others. These devices can be configured to implement specific routing protocols, VLANs, access control policies, and other features to achieve the desired network behavior and performance. Examples of custom network topologies include:

Mesh topology with adaptive routing: This topology can be used in large-scale wireless networks to provide high redundancy and fault tolerance. Adaptive routing protocols such as OLSR or B.A.T.M.A.N. may be used to optimize network performance and reduce congestion.

Hub-and-Spoke topology with VPN: This topology can be used to connect multiple remote offices or branches to a central location using VPN tunnels. A hub router or firewall is used to manage the traffic flow and provide secure connectivity between the spokes.

Cluster topology with load balancing: This topology can be used to create a cluster of web or application servers for high availability. Load balancing devices are used to distribute traffic across multiple servers in the cluster, providing high performance and scalability.

Custom network topologies can offer unique advantages and solve specific problems, but they also require careful design and management to ensure effectiveness and security. Network administrators should consider the specific needs of their organization and consult experienced network designers to create a custom topology that meets these needs.

6.1.4. Random Topology

In computer networking, a random network topology refers to a network topology where connections between nodes are made in a random or stochastic manner. In such a topology, there is no predetermined plan or structure to the connections between nodes. Random network topologies are used in various applications, such as in the

study of social networks, biological networks, and communication networks. They are also used in analyzing network properties, such as connectivity, robustness, and efficiency. It has been shown that they exhibit some interesting and unexpected behaviors, such as the emergence of small-world networks and scale-free networks.

6.1.5. Erdős–Rényi Model

An example of a random network topology is the Erdős–Rényi model. The Erdős–Rényi model, also known as the ER model, is a mathematical model for creating random graphs. Introduced to the field of mathematics by mathematicians Paul Erdős and Alfréd Rényi in 1959, the ER model creates a random graph with " n " nodes starting with " n " isolated nodes and then randomly connecting pairs of nodes with a certain probability " p ". The edges between the nodes are independent and occur with probability " p ". There are two variations of the ER model: the $G(n,m)$ model, which creates a random graph with " n " nodes and m edges, and the $G(n,p)$ model, which creates a random graph with " n " nodes and an edge between each pair of nodes with probability " p ".

The ER model has been used to study various properties of random graphs, including the appearance of the giant component, the phase transition of connectivity, and the degree distribution of the graph. The model has also been applied in various fields such as social networks, computer networks, and biology. However, it should be noted that the ER model assumes a completely random and uniform distribution of edges, which may not always reflect the real structure of many networks. As a result, other network models, such as small-world networks and scale-free networks, have been proposed to better map the properties of real networks. In Figure 8 below we can see a custom random topology is presented [18].

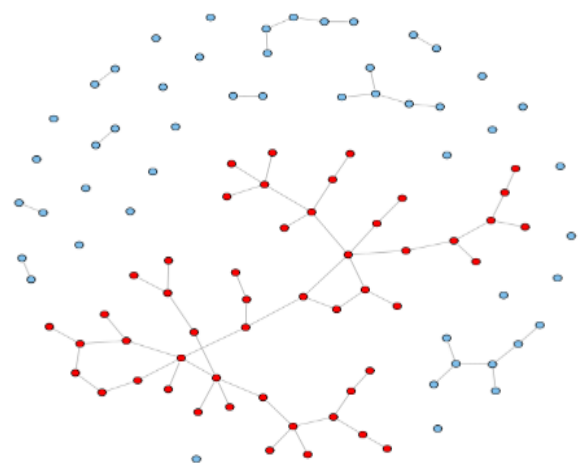


Figure 8: Example of Custom Random Topology using the Erdős–Rényi mathematical model

6.2. Experiment Specifications

The experiments implement topologies based on the random topology which in turn follows the Erdős–Rényi mathematical model. The SDN controller used is POX due to its compatibility with both topologies and the creation and parameterization of topologies through the PYTHON language. The topology creation protocol is OFDP, or otherwise OpenFlow [19].

The experiments examine the following:

- Comparison of system performance according to topologies.
- Comparison of system performance according to topology creation protocol.
- Comparison of system performance according to the number of switches and how the total number of switches affects performance.
- CPU usage.
- RAM usage
- The delay of packet transfer between network nodes.
- The time of creation and destruction of a topology

The above measurements will be compared:

Topologies:

- Linear
- Balanced Tree
- Random Topology

Creation Protocols

- OFDP
- LLDP
- BGP
- LSP
- SNMP
- OVSD

The number of switches will remain steadily increasing, and each switch will be connected to a host in the manner shown in the table below [20].

It is noted that a greater number of switches was achieved than in most similar studies. This fact alone allows for better interpretation of results and is primarily due to the available hardware resources. Table 4 contains the scale of the experiments depending on the number of switches and hosts.

Table 4: Scale of experiments conducted

SWITCHES	HOSTS
2	2

4	4
8	8
16	16
32	32
64	64
128	128
256	256
512	512
1024	1024
2048	2048
4096	4096
8192	8192

6.3. Collection of General Results

In this section, the statistical tables of the data collected from the above experiments will be presented. The controller used is POX and the topology creation protocol is OFDP. It is worth noting that each experiment was performed about a thousand times to ascertain the accuracy percentage of the results, and the deviations were minimal and consistent with the expected pattern. Therefore, the results presented are the overall average. Below the tables of experiment results are presented. Table 5 presents the results using random topologies, Table 6 presents the results using linear topologies and Table 7 presents the results using balanced tree topologies [21].

Table 5: Experiment results using random topologies.

CP U (%)	MEMOR Y (MB)	SWITC H	HOST S	BW (Gbps)	SETU P TIME (sec)	TEAR TIME (sec)
1.9	150	2	2	41	0.092	0.085
2.6	170	4	4	42	0.145	0.136
6.4	210	8	8	48	0.326	0.413
11.2	250	16	16	48	1.256	1.646
13.5	290	32	32	47	2.719	6.167
18.1	330	64	64	38	12.752	22.39
22.4	390	128	128	38	18.393	29.712
27.8	440	256	256	36	26.715	39.513
33.6	625	512	512	33	39.212	58.004
39.2	1100	1024	1024	32	57.454	74.981
44.5	2000	2048	2048	28	83.757	119.046
47.3	3500	4096	4096	26	183.908	244.901
62.9	6800	8192	8192	27	274.483	368.271

Table 6: Experiment results using linear topologies.

CP U (%)	MEMOR Y (MB)	SWITC H	HOST S	BW (Gbps)	SETU P TIME (sec)	TEAR TIME (sec)
1.2	300	2	2	45	0.098	0.067
1.9	340	4	4	44	0.182	0.224
4.8	380	8	8	49	0.295	0.313
9.3	420	16	16	47	0.542	0.621
10.3	480	32	32	48	0.894	1.128
14.5	560	64	64	481	1.889	2.359
19.3	680	128	128	38	3.319	4.858
23.6	1050	256	256	38	6.822	7.254
28.1	1680	512	512	39	14.841	18.952
33.6	2200	1024	1024	37	33.713	39.701
38.4	3500	2048	2048	36	55.915	62.113
42.5	7300	4096	4096	33	98.009	127.989
53.6	12300	8192	8192	31	181.411	229.410

Table 7: Experiment results using balanced tree topologies.

CP U (%)	MEMOR Y (MB)	SWITC H	HOST S	BW (Gbps)	SETU P TIME (sec)	TEAR TIME (sec)
3.2	180	2	2	43	0.150	0.141
4.5	220	4	4	42	0.265	0.181
8.9	270	8	8	38	0.429	0.284
14.7	380	16	16	44	1.854	1.678
17.6	490	32	32	48	3.535	3.280
21.2	600	64	64	41	6.614	7.252
24.8	710	128	128	43	8.325	10.053
29.1	930	256	256	40	17.783	19.993
36.1	1450	512	512	39	26.977	41.900
44.9	2580	1024	1024	41	56.672	77.451
52.4	4310	2048	2048	37	128.334	168.513
59.9	8200	4096	4096	38	190.985	212.717
74.4	15200	8192	8192	30	260.511	332.557

6.3.1. Collection of Latency Results

Latency measurement will be done differently as each network is measured under similar conditions with a fixed packet size, increasing the number of packets and observing how this affects the network. The average and total transfer times are collected. The size of each packet is defined as 1024Bytes (1KB), and simulations will be executed with the corresponding packet numbers [1,10,50,100,500]. In previous studies, a usage limit of about 600 packets of this packet size was observed in Mininet. Below Table 8 is presented in which we can see the latency results of each topology.

Table 8: Latency results of experiments across all topologies.

PACKET NUMBER	TREE AVERAGE LATENCY (ms)	LINEAR AVERAGE LATENCY (ms)	ERDOS RENYI AVERAGE LATENCY (ms)
1	0.048	0.018	0.013

10	0.053	0.027	0.016
50	0.044	0.026	0.018
100	0.031	0.023	0.021
500	0.041	0.015	0.022
TOTAL AVERAGE	0.0434	0.0218	0.0181

7. Analysis of results

The results obtained in the present research are appropriately transformed into diagrams. On the vertical axis, each studied element (CPU, RAM, Bandwidth, Setup Time, Tear Time, Latency) is distributed, while on the horizontal axis there is the number of switches used, thus conclusions are drawn based on the quantity of Switches. In the Latency diagram, on the horizontal axis, the number of Switches is replaced by the number of packets.

7.1. CPU Analysis

The following section provides an in-depth analysis of CPU usage in relation to the number of switches in various network topologies. The data illustrates that CPU usage increases as the number of switches rises. The analysis is based on comparative data from three topologies: balanced tree, random, and linear.

7.1.1. Balanced Tree Topology

Highest CPU Consumption: The balanced tree topology consumes the most CPU resources among the three topologies. This is attributed to the complexity and structure of the tree-branches, which require more processing power to manage the available paths.

CPU Usage Increases with Switches: As the number of switches increases, CPU usage significantly rises. At the peak of 8192 switches, the CPU usage reaches 74.4%, which is 11.5% higher than the random topology and 20.8% higher than the linear topology.

7.1.2. Random Topology:

Close to Balanced Tree: The random topology's CPU consumption is slightly less than the balanced tree but higher than the linear topology. This is due to the adaptable nature of the random topology, which requires complex computations to manage dynamic connections.

Instabilities: Some instabilities in CPU usage are observed, caused by the probability of unsuccessful connections between nodes, which adds variability to the CPU load.

7.1.3. Linear Topology

Least CPU Usage: The linear topology demonstrates the least CPU usage due to its simple and straightforward

connectivity. The simplicity of managing linear connections results in lower processing requirements.

Stable Performance: The linear topology shows stable CPU performance, with less variability and lower overall CPU consumption compared to the other topologies.

Complexity and Resource Allocation: The balanced tree topology requires more CPU resources due to its hierarchical structure. Managing multiple levels and branches in the network involves more processing to maintain efficient routing and data flow. This complexity inherently increases the CPU load as the network scales.

Adaptability of Random Topology: While the random topology is designed for flexibility and adaptability, this also introduces challenges in maintaining stable connections. The CPU must handle dynamic routing and potential connection failures, leading to increased CPU usage and occasional spikes.

Efficiency of Linear Topology: The linear topology benefits from its simplicity, where each switch is directly connected in a straightforward path. This minimizes the processing required for routing decisions, leading to lower and more consistent CPU usage. The linear approach simplifies network management and reduces the computational burden on the CPU.

The analysis highlights that network topology significantly impacts CPU usage. The balanced tree topology, while offering robust and hierarchical structuring, imposes a high CPU load due to its complexity. The random topology, though adaptable, faces challenges with connection stability, leading to variable CPU consumption. Linear topology remains the most efficient in terms of CPU usage, owing to its simple and direct connectivity. These findings are crucial for network administrators and designers, emphasizing the need to consider topology choice based on the expected network load and performance requirements. Balancing complexity, adaptability, and efficiency is key to optimizing network performance and resource utilization. Figure 9 analyses the CPU usage in each experiment.

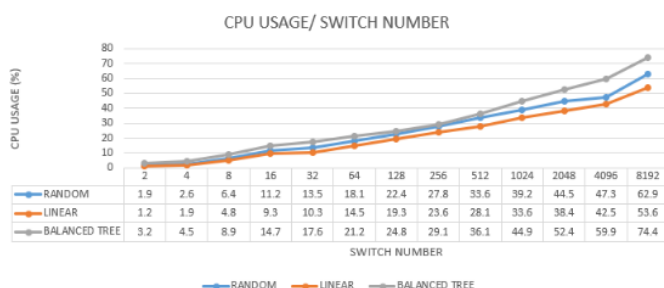


Figure 9: Comparative CPU usage diagram for the three topologies.

7.2. RAM Analysis

The following section provides an in-depth analysis of RAM usage in relation to the number of switches in various network topologies. The data illustrates that RAM usage varies significantly with the topology used and the number of switches in the network. This analysis is based on comparative data from three topologies: linear, balanced tree, and random.

7.2.1. Linear Topology

Initial High Memory Consumption: Initially, the linear topology consumes more RAM compared to the other two topologies, despite using less CPU than the balanced tree. This higher initial memory usage can be attributed to the straightforward but memory-intensive nature of maintaining direct connections between each switch.

Memory Usage Trends: As the number of switches increases, the memory usage grows but at a predictable and steady rate due to the simple structure of the linear topology.

7.2.2. Balanced Tree Topology

High Memory Consumption with Increased Switches: While the number of switches increases, the balanced tree topology eventually consumes the most memory. This is due to the complexity of managing a hierarchical tree structure, which requires more memory to store the state and routing information for multiple levels and branches.

Complexity Impact: The tree-branch structure inherently requires more memory to maintain the hierarchical relationships and efficient routing, resulting in higher memory usage as the network scales.

7.2.3. Random Topology

Lowest Memory Consumption: The random topology consistently shows lower RAM usage compared to the other two topologies. This is largely due to its customization and the retrospective improvements made to its implementation in Mininet, which optimize memory usage.

Efficiency of Customization: Due to its adaptable nature and optimized design, the random topology reduces memory consumption by about 45-55% compared to the linear and balanced tree topologies.

Initial Memory Usage in Linear Topology: The linear topology, despite its simplicity, requires substantial memory initially to establish and maintain direct

connections between each switch. This direct approach, while less CPU-intensive, places a higher initial burden on RAM.

Increasing Complexity in Balanced Tree Topology: As the network grows, the balanced tree topology's memory requirements increase significantly. This is because the hierarchical structure demands more memory to store the details of each level and branch, ensuring efficient data routing and network management.

Optimized Memory Usage in Random Topology: The random topology benefits from its customized and optimized implementation in Mininet. This design reduces unnecessary memory usage and streamlines the management of random connections, leading to significantly lower RAM consumption. The flexibility and adaptability of the random topology also contribute to its efficient memory usage.

The analysis highlights that network topology significantly impacts RAM usage. The linear topology, while simple, initially demands more memory but grows predictably. The balanced tree topology, due to its hierarchical structure, consumes the most memory as the network expands. The random topology, with its optimized and adaptable design, demonstrates the most efficient memory usage. These insights are crucial for network administrators and designers, emphasizing the need to consider topology choice based on the expected network load and performance requirements. Balancing complexity, adaptability, and efficiency is key to optimizing network performance and resource utilization, particularly in terms of memory usage. Figure 10 analyses the RAM usage in each experiment.

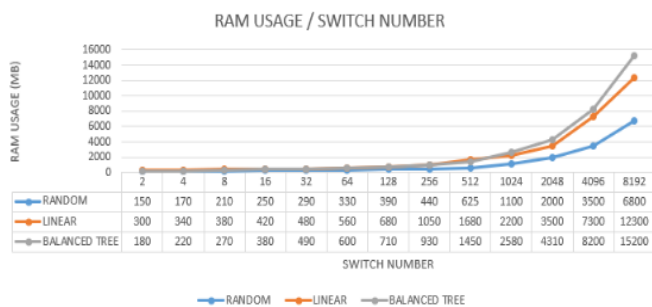


Figure 10: Comparative RAM usage diagram for the three topologies.

7.3. Bandwidth Analysis

The following section provides an in-depth analysis of RAM usage in relation to the number of switches in various network topologies. The data illustrates that RAM usage varies significantly with the topology used and the number of switches in the network. This analysis

is based on the comparative data from three topologies: linear, balanced tree, and random.

7.3.1. Linear Topology

Initial High Memory Consumption: Initially, the linear topology consumes more RAM compared to the other two topologies, despite using less CPU than the balanced tree. This higher initial memory usage can be attributed to the straightforward but memory-intensive nature of maintaining direct connections between each switch.

Memory Usage Trends: As the number of switches increases, the memory usage grows but at a predictable and steady rate due to the simple structure of the linear topology.

7.3.2. Balanced Tree Topology

High Memory Consumption with Increased Switches: While the number of switches increases, the balanced tree topology eventually consumes the most memory. This is due to the complexity of managing a hierarchical tree structure, which requires more memory to store the state and routing information for multiple levels and branches.

Complexity Impact: The tree-branch structure inherently requires more memory to maintain the hierarchical relationships and efficient routing, resulting in higher memory usage as the network scales.

7.3.3. Random Topology

Lowest Memory Consumption: The random topology consistently shows lower RAM usage compared to the other two topologies. This is largely due to its customization and the retrospective improvements made to its implementation in Mininet, which optimize memory usage.

Efficiency of Customization: Due to its adaptable nature and optimized design, the random topology reduces memory consumption by about 45-55% compared to the linear and balanced tree topologies.

Initial Memory Usage in Linear Topology: The linear topology, despite its simplicity, requires substantial memory initially to establish and maintain direct connections between each switch. This direct approach, while less CPU-intensive, places a higher initial burden on RAM.

Increasing Complexity in Balanced Tree Topology: As the network grows, the balanced tree topology's memory requirements increase significantly. This is because the hierarchical structure demands more memory to store the

details of each level and branch, ensuring efficient data routing and network management.

Optimized Memory Usage in Random Topology: The random topology benefits from its customized and optimized implementation in Mininet. This design reduces unnecessary memory usage and streamlines the management of random connections, leading to significantly lower RAM consumption. The flexibility and adaptability of the random topology also contribute to its efficient memory usage.

The analysis highlights that network topology significantly impacts RAM usage. The linear topology, while simple, initially demands more memory but grows predictably. The balanced tree topology, due to its hierarchical structure, consumes the most memory as the network expands. The random topology, with its optimized and adaptable design, demonstrates the most efficient memory usage. These insights are crucial for network administrators and designers, emphasizing the need to consider topology choice based on the expected network load and performance requirements. Balancing complexity, adaptability, and efficiency is key to optimizing network performance and resource utilization, particularly in terms of memory usage. Figure 11 analyses the bandwidth of each experiment.

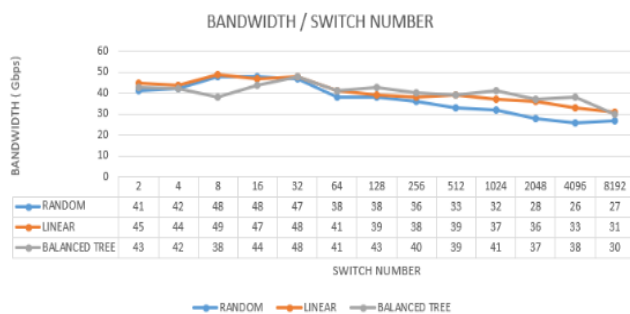


Figure 11: Comparative Bandwidth diagram for the three topologies.

7.4. Setup Time Analysis

The setup time refers to the duration required to create a network topology, measured from the moment the creation command is initiated. The analysis compares the setup times for three different network topologies: random, balanced tree, and linear. The results indicate significant differences in the time taken to establish each topology, highlighting the efficiency and complexity involved in their creation.

7.4.1. Random Topology

Longest Setup Time: The random topology consistently shows the longest time required to create a topology. This is due to its inherent complexity and the need for random

connections between nodes, which involves additional computational overhead to ensure successful creation and connectivity.

Marginally Longer: Among the topologies with long setup times, the random topology takes slightly longer than the balanced tree, indicating higher variability and complexity in establishing random connections.

7.4.2. Balanced Tree Topology

Long Setup Time: The balanced tree topology also exhibits a long setup time, slightly less than the random topology. The hierarchical structure requires careful planning and execution to ensure all branches and levels are correctly established, which adds to the setup time.

Complexity Contribution: The structured nature of the balanced tree, with multiple levels and branches, contributes to the extended time needed for its creation.

7.4.3. Linear Topology

Shortest Setup Time: The linear topology shows a significantly reduced setup time compared to the other two topologies. This is due to its straightforward design, where each node is directly connected to the next in a simple chain.

Efficiency in Large Networks: In very large networks, the linear topology is approximately 30-40% faster to set up than the random and balanced tree topologies. This efficiency is attributed to the minimal complexity in establishing direct connections sequentially.

Complexity and Overhead: The random and balanced tree topologies require more time to create due to their inherent complexity. Random topology involves the creation of non-deterministic connections that need verification and correction, while the balanced tree requires a hierarchical setup with multiple levels, each adding to the overall setup time.

Linear Topology Efficiency: Linear topology's setup process is inherently simpler. Each new node is added in a straightforward manner, reducing the time required for planning and establishing connections. This simplicity translates to a significant reduction in setup time, especially as the network scales.

Scalability and Performance: As the network size increases, the difference in setup times becomes more pronounced. The linear topology's efficient setup process becomes increasingly advantageous in larger networks, where the time savings are substantial compared to the more complex topologies.

Practical Implications: For practical applications, the choice of topology can significantly impact the time required to deploy a network. In scenarios where rapid deployment is critical, the linear topology offers a clear advantage. Conversely, if the network's structural complexity and adaptability are priorities, the additional setup time for random or balanced tree topologies may be justified.

The setup time analysis underscores the importance of topology selection based on deployment time requirements and network complexity. The linear topology offers the fastest setup time, making it suitable for scenarios requiring quick deployment and straightforward management. The random and balanced tree topologies, while taking longer to set up, provide more complex and potentially more resilient network structures. Understanding these trade-offs is essential for network administrators and designers to optimize network deployment strategies and achieve the desired balance between setup efficiency and structural complexity. Figure 12 show the results of setup time of the experiments.

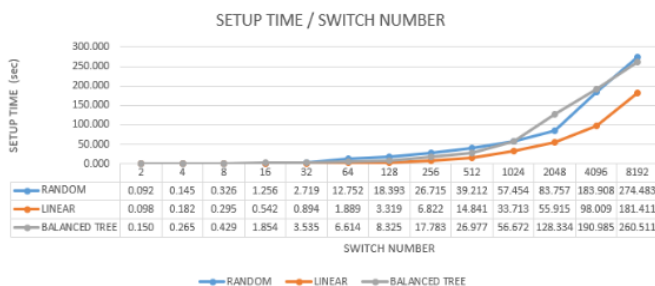


Figure 12: Comparative setup time diagram for the three topologies

7.5. Tear Time Analysis

The tear time refers to the duration required to dismantle a network topology, measured from the moment the destruction command is initiated. The analysis compares the tear times for three different network topologies: linear, balanced tree, and random. The results indicate significant differences in the time taken to dismantle each topology, highlighting the efficiency and complexity involved in their destruction.

7.5.1. Linear Topology

Shortest Tear Time: As expected, the linear topology takes the least amount of time to tear down. This is due to its straightforward structure, where nodes are connected in a simple chain, making it easy to dismantle.

Efficiency in Large Networks: In large networks, the linear topology shows about 30-40% faster tear times compared to the random and balanced tree topologies.

This efficiency is attributed to the minimal complexity involved in breaking the direct sequential connections.

7.5.2. Random Topology

Longest Tear Time: The random topology consistently shows the longest tear time among the three topologies. This is due to the complexity and unpredictability of its connections, which require additional time to ensure all links are properly dismantled.

Peak Tear Times: The tear time peaks higher in the random topology, reflecting the inherent variability and instability in its structure.

7.5.3. Balanced Tree Topology

Long Tear Time: The balanced tree topology also exhibits a long tear time, like the random topology but slightly less. The hierarchical structure requires careful dismantling of multiple levels and branches, adding to the overall tear time.

Deviations in Linearity: Some deviations in the linearity of tear time are observed in the balanced tree topology. These deviations are due to the changes in the tree structure as different branches and levels are dismantled.

Efficiency of Linear Topology: The linear topology's simplicity extends to its tear-down process. Each node is directly connected to its predecessor and successor, making it easy to break these connections in sequence. This straightforward dismantling process results in consistently lower tear times.

Complexity in Random Topology: The random topology's longer tear time is attributed to its complex and unpredictable nature. The random connections between nodes mean that each dismantling process is unique and requires more time to ensure all links are effectively broken. This variability results in higher and more inconsistent tear times.

Structured Dismantling in Balanced Tree Topology: The balanced tree topology requires careful dismantling of its hierarchical structure. Each branch and level must be carefully broken down, which increases the overall tear time. The deviations in linearity are due to the varying complexity of dismantling different parts of the tree.

Instabilities and Variability: Both the random and balanced tree topologies show instabilities and variability in tear times. These instabilities are natural given the complexity of the structures and the need for careful dismantling to avoid leaving residual connections.

The tear time analysis highlights the impact of network topology on the efficiency of dismantling processes. Linear topology offers the fastest and most efficient tear-down times, making it suitable for scenarios requiring quick and straightforward network reconfiguration. The random and balanced tree topologies, while offering more complex and potentially more resilient structures, require significantly more time to dismantle. Understanding these differences is crucial for network administrators and designers in optimizing network management strategies, particularly in environments where frequent reconfiguration is necessary. Figure 13 show the results of tear time for each experiment.

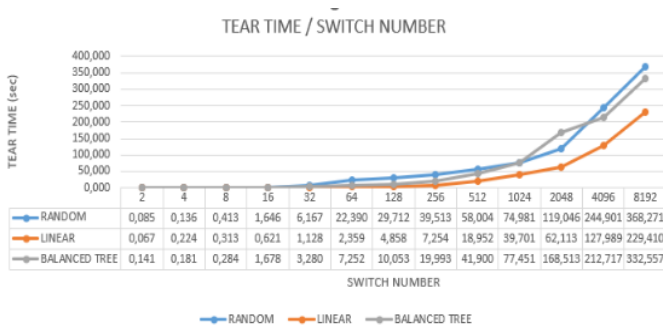


Figure 13: Comparative tear time diagram for the three topologies.

7.6. Latency Analysis

Latency measurement refers to the time taken for an information packet to travel from one network node to another, measured in milliseconds (ms). For this analysis, the packet size was set to 1024 Bytes (1 Kilobyte), and the maximum number of transferred packets was capped at 500, a limit identified in previous Mininet research for reliable measurements.

7.6.1. Balanced Tree Topology

7.6.1.1. Highest Delay

The balanced tree topology exhibits the highest latency among the three topologies. This significant delay is due to the complexity of its hierarchical structure, which requires packets to traverse multiple levels and branches before reaching their destination.

7.6.1.2. Impact of Complexity

The structured nature of the balanced tree increases the distance and processing time for packets, leading to higher latency.

7.6.2. Linear and Random Topologies

7.6.2.1. Similar Delays

Both linear and random topologies show similar latency measurements, but still lower than the balanced tree

topology. These topologies have less complex routing paths, which reduces the overall transmission time.

7.6.2.2. Comparative Analysis

Although their delays are similar, the linear topology generally maintains a slightly more predictable and stable latency due to its straightforward path structure, while the random topology may experience more variability due to its non-deterministic connections.

7.6.3. Latency Comparison

Double the Delay in Balanced Tree: The latency in the balanced tree topology is at least double that of the other two topologies. This stark difference underscores the impact of hierarchical complexity on network performance.

7.6.4. Balanced Tree Topology

7.6.4.1. Hierarchical Routing

The balanced tree's multi-level structure means that packets often need to travel through several intermediary nodes (branches) before reaching their target. Each additional hop adds to the overall delay, resulting in the highest latency.

7.6.4.2. Increased Processing Time

Managing and routing through the hierarchical levels introduces additional processing delays, further contributing to the higher latency.

7.6.5. Linear Topology

7.6.5.1. Direct Pathways

The linear topology benefits from direct, sequential connections between nodes. This straightforward routing minimizes the number of hops and processing required, leading to more predictable and lower latency.

7.6.5.2. Stable Performance

The linear nature of the topology ensures consistent performance, with each packet following a clear and defined path.

7.6.6. Random Topology

Variable Pathways: The random topology features non-deterministic connections, meaning packets may traverse different paths depending on the network state. This variability can introduce occasional increases in latency, although the average delay remains lower than the balanced tree topology.

Adaptability and Efficiency: Despite its variability, the random topology's design aims to balance load and optimize pathways, helping to maintain relatively low latency overall.

The latency analysis highlights the significant influence of network topology on transmission delays. The balanced tree topology, with its complex hierarchical structure, results in the highest latency, making it less suitable for applications requiring rapid data transmission. Linear topology, with its direct and predictable pathways, offers the lowest latency and stable performance, ideal for time-sensitive applications. The random topology, while variable, maintains lower latency than the balanced tree and can adapt to different network conditions effectively. These insights are crucial for network administrators and designers to optimize network performance based on specific latency requirements and application needs. Figure 14 presents the latency results.

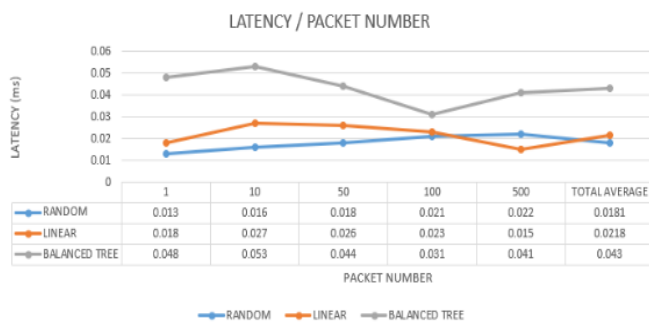


Figure 14: Comparative latency diagram for the three topologies.

7.7. Future Research

The content of this specific postgraduate work is a fundamental pillar of research on SDN networks and extends existing research in the field of computer science, networks, and telecommunications. Future research could be expanded on an even larger scale with the aid of supercomputers from major academic structures to show how a total shift in networking towards SDN would affect the internet and the world in general. With the right available resources, even more realistic simulations would be possible, aiming for direct integration, improvement, and gradual adaptation, initially in academic structures and subsequently in society, aiming for a stronger global network that would be more efficient, reliable, and capable of withstanding the continuously increasing needs of modern society. Lastly, as an extension of what was studied, the combination of currently active protocols to create a new improved one is feasible.

8. Conclusions

8.1. Performance of Topologies

Through experimental procedures, we can understand how SDN functions best and the operation of distinct topologies. Large-scale networks are created, and their characteristics are studied. This postgraduate work achieves an understanding of these network structures in real-time and how their effective application is possible in real-time.

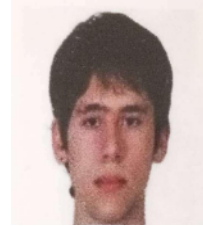
The results indicate that the balanced tree topology consumes the most CPU resources due to its complexity, followed by the random topology. Linear topologies showed the least CPU usage. RAM consumption was highest in the balanced tree topology, while the random topology demonstrated lower RAM usage due to its customized nature. Latency measurements revealed that the balanced tree topology had the highest delay, while the linear and random topologies performed better with less delay. The random topology achieves improved results due to its adaptability and the ability to be parameterized. However, there is always the possibility of unsuccessful connections in the random topology, which affects performance but adds a more "realistic" application. The linear topology remains simple and maintains top performance. In contrast, the balanced tree topology, due to its architecture, reduces performance as it is burdened and expanded because of the complexity and calculations required for the successful creation of the tree. The use of the POX controller in collaboration with the OFDP protocol facilitated the expansion of SDN network sizes through parameterization.

References

- [1] B. A. Nunes, M. Mendonca, N. Nguyen, K. Obraczka and T. Turtletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617-1634, Third Quarter 2014, doi: 10.1109/SURV.2014.012214.00180.
- [2] N. V. Oikonomou, S. V. Margariti, E. Stergiou, and D. Liarokapis, "Performance Evaluation of Software-Defined Networking Implemented on Various Network Topologies," in *2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, Preveza, Greece, 2021, pp. 1-6, doi: 10.1109/SEEDA-CECNSM53056.2021.9566213.
- [3] A. Zacharis, S. V. Margariti, E. Stergiou, and C. Angelis, "Performance evaluation of topology discovery protocols in software defined networks," in *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Heraklion, Greece, 2021, 135-140, doi: 10.1109/NFV-SDN53031.2021.9665006.
- [4] M. Guo and P. Bhattacharya, "Controller Placement for Improving Resilience of Software-Defined Networks," in *2013 Fourth International Conference on Networking and Distributed*

- Computing, Los Angeles, CA, USA, 2013, 23-27, doi: 10.1109/ICNDC.2013.15.
- [5] IETF RFC 7426, "Request for Comments: 7426, ISSN: 2070-1721 EICT. Category: Informational," K. Pentikousis, Ed., 2015.,doi:10.20535/2411-2976.12021.24-32
- [6] D. Kreutz, F. M. V. Ramos, P. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," arXiv, 2014, [Online], doi:10.48550/arXiv.1406.0440
- [7] A. Nayak, A. Reimers, N. Feamster, and R. Clark, "Resonance: Dynamic access control in enterprise networks," in Proc. Workshop: Research on Enterprise Networking, Barcelona, Spain, 2009, 1-6, doi:10.1145/2602204.2602219
- [8] A. Voellmy and P. Hudak, "Nettle: Functional reactive programming of OpenFlow networks," in Proc. Workshop on Practical Issues in Programming, 2009, pp. 1-6,doi: 10112206
- [9] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, "ElasticTree: Saving energy in data center networks," in Proc. 7th USENIX Symposium on Networked Systems Design and Implementation, 2010, 1-6,doi: 10.5555/1855711.1855728
- [10] R. Wang, D. Butnariu, and J. Rexford, "OpenFlow-based server load balancing gone wild," in Hot-ICE, 2011, 1-6,doi: 10.5555/1972422.1972438
- [11] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, and A. Vahdat, "B4: Experience with a globally deployed software defined WAN," in ACM SIGCOMM, 2013, pp. 1-6,doi: 10.1145/2534169.2486019
- [12] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, and S. Shenker, "Onix: A distributed control platform for large-scale production networks," in OSDI, vol. 10, 1-6, 2010,doi: 10:351-364
- [13] X. Zhao, S. S. Band, S. Elnaffar, M. Sookhak, A. Mosavi, and E. Salwana, "The Implementation of Border Gateway Protocol Using Software-Defined Networks: A Systematic Literature Review," IEEE Access, vol. 9, 112596-112606, 2021, doi: 10.1109/ACCESS.2021.3103241.
- [14] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, "A roadmap for traffic engineering in SDN-OpenFlow networks," Elsevier Computer Networks, vol. 71, pp. 1-30, 2014,doi:10.1016/j.comnet.2014.06.002
- [15] ONF TR-537, "Negotiable Datapath Model and Table Type Pattern Signing," Version 1.0, Sep. 2016, pp. 1-6.
- [16] N. Handigol, M. Flajslik, S. Seetharaman, N. McKeown, and R. Johari, "Aster*x: Load-balancing as a network primitive," in ACLD '10: Architectural Concerns in Large Datacenters, 2010, 1-6,doi: 10.1109/GREE.2014.9
- [17] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McKeown, and G. Parulkar, "Can the production network be the testbed?" in Proc. 9th USENIX OSDI, Vancouver, Canada, 2010, 1-6, doi: 10.5555/1924943.19249691
- [18] A. Rodriguez-Natal, M. Portoles-Comeras, V. Ermagan, D. Lewis, D. Farinacci, F. Maino, and A. Cabello, "LISP: a southbound SDN protocol?" IEEE Communications Magazine, vol. 53, 201-207, 2015, doi: 10.1109/MCOM.2015.7158286.
- [19] K. Greene, "TR10: Software-defined networking," MIT Technology Review, 2009, [Online]. Available: <http://www2.technologyreview.com/article/412194/tr-10-software-defined-networking/>, doi:10.1109/COMST.2016.2633579
- [20] M. Casado, M. J. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker, "Ethane: Taking control of the enterprise," in ACM SIGCOMM '07, 2007, 1-6, doi:10.1145/1282427.1282382
- [21] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, "NOX: Towards an operating system for networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 3, 105-110, 2008, doi:10.1145/1384609.1384625

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



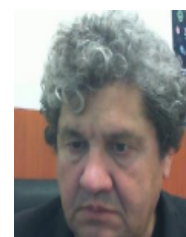
NIKOLAOS V. OIKONOMOU has received his BSc degree from the University of Ioannina, Department of Informatics and Telecommunications in 2021. He received his MSc degree from the same institution in 2023. He is an academic researcher also

working as a private tutor in the field of Computer Science and mathematics. He has years of experience as a Computer Engineer, IT specialist and Network consultant. He also taught at the University of Ioannina and worked as an application developer.



DIMITRIOS V. OIKONOMOU has received his BSc from University of Western Macedonia, Department of Regional and Cross Border Studies in 2024. He is currently an active research member of the University of Western Macedonia and is about to begin his MSc

studies.



ELEFThERIOS STERGIOU has received his Meng Degree from National University of Athens, Department of Electrical Engineering and Informatics. He received his MSc in Telematics from the University of Sheffield Department of Computer




Science and his PhD degree in the field of Computer Science from University of Patras. Currently he is an Associate Professor at the Department of Informatics and Telecommunications, University of Ioannina. His research interests include mainly Performance issues of computer networks and Telecommunication systems.



DIMITRIOS LIAROKAPIS has received his Meng Degree from University of Patras, Department of Computer Engineering and Informatics. He received his MSc and PhD degree in the field of Computer Science from the University of Massachusetts Boston. Currently he is a professor of

practice Professor at Department of Informatics and Telecommunications, University of Ioannina since 2012. His research interests include mainly databases and programming languages.

Keratoconus Disease Prediction by Utilizing Feature-Based Recurrent Neural Network

Saja Hassan Musa¹ , Qaderiya Jaafar Mohammed Alhaidar¹ , Mohammad Mahdi Borhan Elmi² 

¹Department of Electrical Engineering University of Islamic Azad University, Isfahan (Khorasgan) Branch, Isfahan, Iran

²Department of Electrical and Electronic Engineering Faculty of Istanbul Aydin University, Istanbul, Turkey

*Corresponding author: Mohammad Mahdi Borhan Elmi, Istanbul Aydin University, mmahdielmi@stu.aydin.edu.tr

ABSTRACT: Keratoconus is a noninflammatory disorder marked by gradual corneal thinning, distortion, and scarring. Vision is significantly distorted in advanced case, so an accurate diagnosis in early stages has a great importance and avoid complications after the refractive surgery. In this project, a novel approach for detecting Keratoconus from clinical images was presented. In this regard, 900 images of Cornea were used and seven morphological features consist of area, majoraxislength, minoraxislength, convexarea, perimeter, eccentricity and extent are defined. For reducing the high dimensionality datasets without deteriorate the information significantly, principal component analysis (PCA) as a powerful tool was used and the contribution of different PCs are determined. In this regard, Box plot, Covariance matrix, Pair plot, Scree Plot and Pareto plot were used for realizing the relation between different features. Improved recurrent neural network (RNN) with Grey Wolf optimization method was used for classification. Based on the obtained results, the average prediction error of the visual characteristics of a patient with keratoconus six and twelve months after the Kraring ring implantation using RNN are 9.82% and 9.29%, respectively. The average error of estimating characteristics of predicting the visual characteristics of a patient with keratoconus six and twelve months after myoring ring implantation are 11.46% and 7.47% respectively.

KEYWORDS: Cornea disease, Feature extraction, Keratoconus prediction.

1. Introduction

The cornea is the outer layer of the eye, so the structural and repair properties of the cornea are essential for protecting the inner contents, maintaining the shape of the eye and achieving light refraction. Keratoconus is a noninflammatory disorder marked by gradual corneal thinning, distortion, scarring, along with increasing corneal high-order aberrations [1]. Keratoconus' pathogenic mechanisms have been studied for a very long time. The course of the disease is evidenced by a loss of visual acuity that cannot be compensated by using glasses. Corneal thinning commonly precedes ectasia and diagnosing keratoconus is part of the preoperative examination of patients undergoing corneal refractive surgery (LASIK, PRK, SMILE) and some intraocular surgery patients who desire optimal refractive outcomes (such as premium intraocular lenses for cataract surgery, secondary phakic intraocular lenses, etc) [2]. The function

of the cornea is to clarify the front surface of the eye, which the keratoconus disease changes it into a cone shape (Figure 1). Traditionally, corneal topography, which measures the anterior curvature of the eye, was used to detect Keratoconus. In the early stages of keratoconus and without clinical signs, diagnosis of disease is difficult [3].

In recent years, corneal biomechanics (the corneal response to stress and the cornea's ability to resist deformation/distortion) is being used to diagnose patients with corneal ectatic disorders (such as keratoconus) because these conditions are characterized by inherent weaknesses in the cornea's biomechanical properties. Keratoconus (KTC) is detected in 1 out of every 2000 people in general [4].

In keratoconus, corneal pressure is no longer maintained by the cornea due to structural abnormalities produced by corneal thinning. Consequently, the cornea

deforms into a conical shape. Vision is significantly distorted in advanced case, so an accurate diagnosis in early stages has a great importance and avoid complications after the refractive surgery. In the advanced stages of this disease, the cornea completely changes shape and the patient needs a cornea transplant.

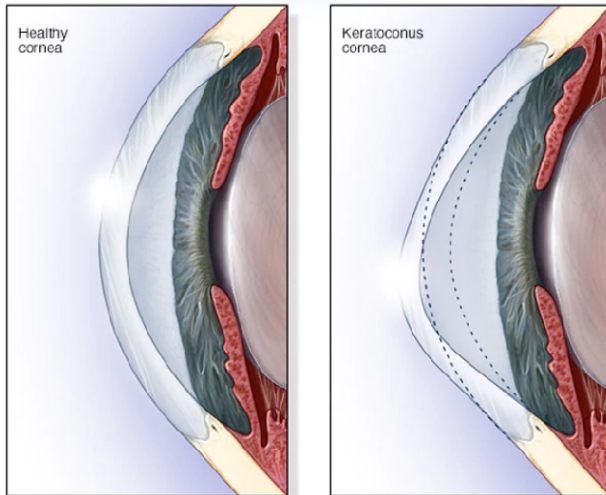


Figure 1: Normal eye (a) and keratoconus eye (b) [5].

Thanks to the technical advances in image processing, it is possible to diagnose this disease in a reliable and non-invasive way. Typically keratoconus specialists using manual assessment to evaluate different corneal attributes and sometimes embrace machine learning techniques to predict the presence of disease [6]. In other side, corneal topography is prepared by devices such as Pentacam and Erbscan. The Pentacam device is one of the most advanced corneal imaging devices, which not only provides very accurate images of the cornea in a very short time, but also enables the examination and analysis of the characteristics of other interior parts. Usually, on the first page of the pentacam report, four cornea maps are placed next to each other (Figure 2). These four maps are:

1. Corneal curvature map
2. Height map of the anterior surface of the cornea
3. Height map of the posterior surface of the cornea
4. Corneal thickness map [7].

In [8], the authors extracted features using four Pentacam-generated refractive maps and retrieved 12 features from each map based on particular diameters. They built a dataset consisting of 40 patients in total and used 30 patients for training and 10 for testing, then obtained results with 90% accuracy. In [9], support vector machine classifier was used, 22 features were extracted from Pentacam and their dataset consisted of 860 patients. They used 10-fold cross validation to train and validate their models and obtained cross-validation accuracies of 98.9%, 93.1% and 88.8% for three different classifiers.

In [10], a 2-way classifier for distinguishing between early stage Keratoconus and normal eyes was presented. The writers compared 25 machine learning models and achieved a test accuracy of 94% using 8 features and a dataset of 3151 patients. Writers of [11] proposed a deep learning approach to classify between Keratoconus and normal eyes. They used data from 304 Keratoconus and 239 normal eyes and reported an accuracy of 0.991 in classifying. In [12], an innovative neural network structure is used. In the designed approach, image and clinical risk factor data are applied as an input data to the multilayer perceptron layer to estimate the disease and in order to improve the results, reinforcement learning techniques have also been evaluated. With considering the potential of severe surgical infections, special precautions must be observed [13].

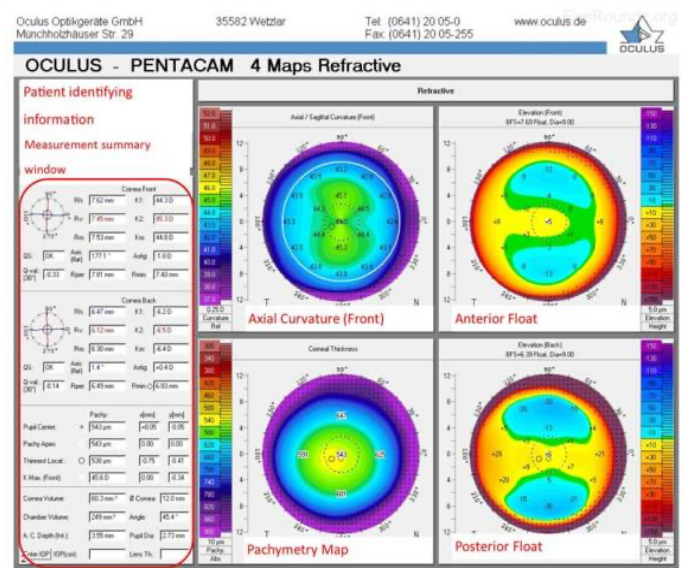


Figure 2: An example of images recorded from the cornea by the Pentacam device [4].

In Table. 1 the review of comparison between different method are shown. From the review of the articles, it can be achieved that the combination of recurrent neural network with feature extraction classification methods has not been used to identify the disease through medical image processing. The proposed approach not only has proper accuracy, but it is also able to be used well for patients who have passed a long time since their kaeraring and myoring operations and has high flexibility in identifying the disease in various conditions. The main contributions of this paper are as follows:

- Combine the recurrent neural network with PCA-based feature selection methods for improving the efficiency of disease identification.
- Applying different classification methods such as logistic regression (IR), k-nearest neighbors (KNN) and decision tree (DT) on dataset and comparing the results.

- Using many criterias such as Confusion Matrix, Decision Boundary, Prediction error and F-1 score for extracting the main features of images.
- Evaluating the presented approach on data set which is gathered after keraring and myoring operations.

Table 1: Comparison between different results of literatures

Ref.	Dataset (images)	Approach	Accuracy
[9]	3510	SVM	> 98%
[11]	546	CNN	> 99%
[12]	570	MLP	> 83%
[14]	2140	AIN	> 92%
[15]	3395	CNN	> 78%
[16]	857	FNN	> 96%
[17]	510	RF	> 75%

In order to explain the proposed approach and present the obtained results, the structure of this article continues as follows. In the next part, feature selection methodology and it's importance for classification algorithms is discussed. Then the steps of applying principal component analysis method on extracted data are presented. After that, characteristics of recurrent neural network are described. In the fifth section, objective function as well as useful benchmarks in this project are formulated. Then, components of data set are defined. Next simulation results after applying PCA and Shapley analysis methods are presented. In this section, four different scenarios for keratoconus classifications after keraring and myoring operations are defined and their results are compared to each other. Before the conclusion section, comparison between different approaches are done for analyzing the efficiency of proposed approach.

2. Basic Concepts

2.1. Feature Selection

Feature selection method is a powerful tool in machine learning and has a great role in clinical intelligence operations. Having a lots of processing capacity and spending a long time to work on the data sometimes is not enough to extract useful information, so it is necessary to use appropriate alternatives to operate with this dataset. There are four important reasons why feature selection is essential:

- I. Spare the model to reduce the number of parameters.
- II. Decrease the learning time.
- III. Reduce overfilling by improving the generalization.
- IV. Avoid the problems of dimensionality [18].

One of the effective solutions to improve the processing efficiency with considering the quality of the results at the

proper level, is using principal component analysis (PCA). PCA help to reduce the size of the huge dataset while considering the original features. In this method, by removing many of the obtained features, the average accuracy of the results can be kept more than 70%.

2.2. Principal component analysis

Suppose there is a data set in the form of a matrix X with dimensions $n \times p$. In order to implement the PCA method, four steps are defined:

2.2.1. Standardization

In order to normalize, average value for each column is calculated according to (1). Then the central matrix is obtained according to the (2).

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (1)$$

$$Y = H.X \quad (2)$$

Here: \bar{x} – is a mean vector;

Y – is a centered data matrix;

X – is a centring matrix.

2.2.2. Covariance matrix computation

Sometimes, the variables are very close to each other, so in these situations, by removing similar ones, it is possible to improve the processing efficiency without causing a significant decreasing in the accuracy. This matrix is calculated as (3) and S is a covariance matrix.

$$S = \frac{1}{n-1} Y^T Y \quad (3)$$

Here: S – is a covariance matrix.

2.2.3. Eigen decomposition

In this step, the eigenvalues and eigenvectors of S are calculated. The obtained results show the direction and variation of each principal components (PCs) (Equation (4)). In this equation, A is a orthogonal matrix of eigenvectors and Λ represents diagonal matrix of eigenvalues.

$$S = A.\Lambda.A^T \quad (4)$$

where: A – is a orthogonal matrix of eigenvectors;

Λ – is a diagonal matrix of eigenvalues.

2.2.4. Principal Components

In the last step, transfer matrix (Z) is calculated which rows represent observations and columns represent PCs (Equation (5)).

$$Z = Y.A \quad (5)$$

2.3. Recurrent Neural Network

The artificial neural network (ANN) technique is a computer-based approach that does not require an estimator to model the system under study [19]. In this method, the patterns are constructed from previous system data and then a mapping is created between the input variables and desired outputs, which is then estimated in order to perform the estimation. In the last decade, neural networks have been made up of multiple layers, known as recurrent neural networks (RNNs). RNNs are a family of algorithms that can include dependency factors between successive time steps. However, as demonstrated by writers of [20], the proposed approach suffers from a gradient divergence pattern that can make factor participation difficult in the long run.

It must be noted that the use of long short-term memory units (LSTM) could be a solution to overcome the problem of gradient divergence. In other side, A RNN is a type of ANN that simulates discrete-time dynamic systems. The structure of such a network is expressed by (6) and (7) [21]. Having a set of N sequences of training data (Equation (8)), RNN parameters are estimated by minimizing the cost function (Equation (9)). RNN parameters should be calculated by a technique such as stochastic gradient descending (SGD) algorithm and by taking the gradient from the cost function expressed in (10) [22]. In the standard RNN structure, the range of accessible content is very limited in practice.

$$h_t = f_h(x_t, h_{t-1}) \quad (6)$$

$$y_t = f_o(h_t) \quad (7)$$

$$D = \{(x_1^{(n)}, y_1^{(n)}), \dots, (x_{Tn}^{(n)}, y_{Tn}^{(n)})\}_{n=1}^N \quad (8)$$

$$J(\theta) = \frac{1}{N} \sum_{n=1}^N \sum_{t=1}^{Tn} d(y_t^{(n)}, f_o(h_t^{(n)})) \quad (9)$$

$$h_t^{(n)} = f_h(x_t^{(n)}, h_{t-1}^{(n)}) \quad (10)$$

Here: t – is time;

x_t – is input;

h_t – is hidden state;

y_t – is output;

f_h – is state transition function;

f_o – is output function.

The problem is that the influence of a given input on the hidden layer and thus on the entire network decays exponentially and vanishes. This problem is known as gradient fading. In 1977, the long short-term memory (LSTM) neural network was introduced by Hackertier and Schmidber, in which the neurons of the hidden layer were replaced by memory blocks, and thus the challenge of long sequences was solved. An LSTM network is similar to a standard RNN except that the summation units (internal value) of the neurons in the hidden layer are replaced by memory blocks.

3. Objective Function

The primary purpose of this study is to classify the normal and disease eyes in correct groups. In this regard, classification is performed on the different varieties of Keratoconus images. This dataset consists of 900 images in two different categories (450 images for Keratoconus and 450 images for normal eyes). The classification is conducted following a selection of essential features using feature selection algorithms. For this purpose, consider a dataset D with N attributes, where each tuple is composed of N values. For each new tuple X , classifier predict that $X \in C_i$ if the class C_i has a highest probability condition on X . In several classification phases for keratoconus, the performance of generated findings is evaluated based on classification accuracy, recall, f1-score, ROC curve, and prediction time. Precision is a metric that reflects how accurate is the model, i.e., how many of them are actually positive [23]. In this paper, precision represents the proportion of eyes for which keratoconus was accurately predicted by (11). The recall is the measure which identify true positives (Equation (12)). Thus, for all instances which actually have keratoconus disease, recall calculates how many samples correctly realized as having keratoconus disease. The F1-score is a metric combining false positives and negatives to make a balance between precision and recall [23]. It is a weighted average of the precision and recall. Model is regarded true when the F1-score is 1, and false when the F1-score is 0 (Equation (13)). Finally, accuracy is a common metric for describing the classification performance of a model across all classes. It reflects the proportion of accurate forecasts relative to the total number of predictions (Equation (14)). Flowchart of the proposed approach is presented as Figure 3.

$$Precision = \frac{TP}{TP + FP} \quad (11)$$

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (13)$$

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \quad (14)$$

Here: TP – are True Positives;

FP– are False Positives;

FN – are False Negatives;

ACC– is acceleration index;

TN– are True Negatives.

4. Results Discussion

4.1. Dataset

In this regard, 7 morphological features are defines as Table. 2. In order to classify the different varieties of Keratoconus images, the dataset consist of 900 images is considered for this study. The minimum, mean, maximum and standard deviation of morphological features for both types of corneas are given in Table 3.

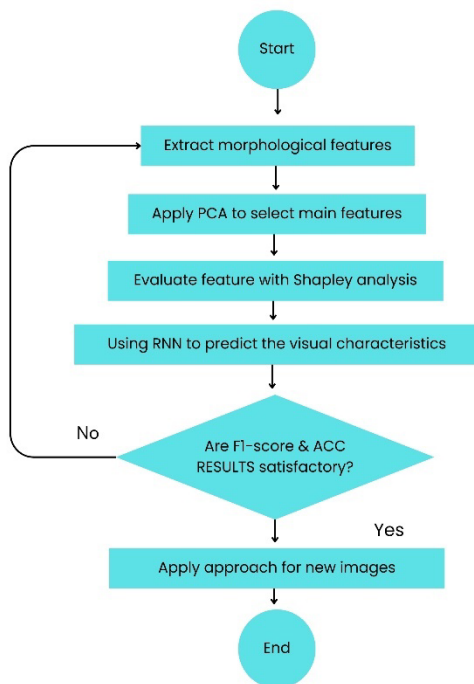


Figure 3: Flowchart of proposed approach

4.2. PCA-based feature analysis

After extraction of features, PCA is applied to reduce the amount of features, while keeping the almost all of important information from dataset. This method extracts variables which are called principal components (PCs) that are uncorrelated to each other. The obtained result is shown in Figure 4. With using PCA on cornea dataset, The matrix of eigenvectors and eigenvalues are obtained as (15) and (16), respectively.

Table 2: Morphological Features

Feature	Definition
Area	No. of pixel's boundaries
Major Axis Length	Length of longest line on the Keratoconus image

Minor Axis Length	Length of shortest line on the Keratoconus image
Eccentricity	Ellipse's eccentricity with the equal moments
Convex Area	No. of pixels of the smallest convex cornea
Extent	Keratoconus' region vs. total pixels of bounding box
Perimeter	Distance between boundaries & pixels around of cornea

Table 3: Statistical data of different features

Feature	Min.	Mean	Max.	Standard Deviation
Area	25387	87804.1	235047	39002.11
Major Axis Length	225.63	430.93	997.292	116.035
Minor Axis Length	143.71	254.488	492.275	49.989
Eccentricity	0.349	0.782	0.962	0.09
Convex Area	26139	91186.1	278217	40769.29
Extent	0.38	0.7	0.835	0.053
Perimeter	619.07	1165.90	2697.76	273.764

$$A = \begin{bmatrix} 0.448 & -0.116 & 0.005 & -0.111 & -0.611 & -0.100 & -0.624 \\ 0.443 & 0.137 & -0.101 & 0.495 & 0.0876 & -0.686 & 0.228 \\ 0.389 & -0.375 & 0.236 & -0.656 & 0.384 & -0.240 & 0.130 \\ 0.203 & 0.611 & -0.629 & -0.426 & 0.075 & 0.054 & 0.020 \\ 0.451 & -0.0877 & 0.037 & 0.056 & -0.392 & 0.471 & 0.640 \\ -0.056 & -0.667 & -0.731 & 0.109 & 0.057 & 0.023 & -0.002 \\ 0.451 & 0.034 & 0.044 & 0.340 & 0.555 & 0.487 & -0.364 \end{bmatrix} \quad (15)$$

$$\lambda = \begin{bmatrix} 4.838 \\ 1.455 \\ 0.630 \\ 0.057 \\ 0.022 \\ 0.006 \\ 0.001 \end{bmatrix} \quad (16)$$

$$V_j = \frac{\lambda_j}{\sum_{j=1}^p \lambda_j} \quad , \quad j = 1, 2, \dots, p \quad (17)$$

$$V_1 = 0.690 \quad V_2 = 0.2076 \quad V_3 = 0.090 \quad V_4 = 0.008 \quad (18)$$

$$V_5 = 0.003 \quad V_6 = 0.001 \quad V_7 = 0.000$$

$$Z_1 = 0.448 X_1 + 0.443 X_2 + 0.389 X_3 + 0.203 X_4 + 0.451 X_5 - 0.056 X_6 + 0.451 X_7 \quad (19)$$

$$Z_2 = -0.116 X_1 + 0.137 X_2 - 0.375 X_3 + 0.611 X_4 - 0.088 X_5 - 0.667 X_6 + 0.034 X_7 \quad (20)$$

Obtained results for this part is represented in Table 4. After applying PCA, the best result belonged to KNN classifier with accuracy equal to 86.04%. According to results extracted by applying PCA, if the dimension of features reduced from $r = 7$ to $r = 2$ then the calculated error was less than 0.5%.

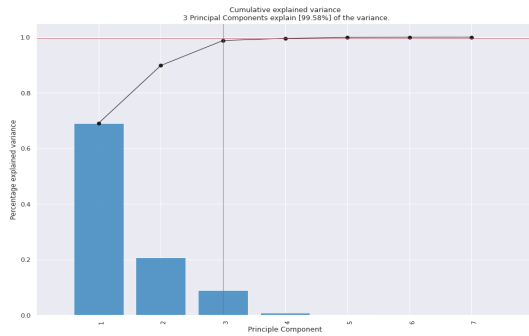


Figure 4: Pareto Plot

Table 4: Classification models and their statistics after PCA

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	NCC	TT (Sec)
knn	K Neighbors Classifier	0.8604	0.9060	0.8108	0.9032	0.8541	0.7211	0.7255	0.014
lr	Logistic Regression	0.8570	0.9206	0.8424	0.8748	0.8561	0.7139	0.7179	0.012
ridge	Ridge Classifier	0.8569	0.0000	0.7969	0.9102	0.8486	0.7142	0.7212	0.008
lda	Linear Discriminant Analysis	0.8569	0.9213	0.7969	0.9102	0.8486	0.7142	0.7212	0.012
lightgbm	Light Gradient Boosting Machine	0.8553	0.9056	0.8283	0.8792	0.8517	0.7107	0.7137	0.207
nb	Naive Bayes	0.8552	0.9199	0.7933	0.9089	0.8467	0.7107	0.7171	0.013
ada	Ada Boost Classifier	0.8517	0.9077	0.8145	0.8839	0.8485	0.7036	0.7076	0.079
qda	Quadratic Discriminant Analysis	0.8463	0.9112	0.7865	0.8994	0.8370	0.6930	0.7011	0.011
rf	Random Forest Classifier	0.8429	0.9069	0.8004	0.8800	0.8366	0.6859	0.6909	0.185
gbc	Gradient Boosting Classifier	0.8412	0.9094	0.8144	0.8657	0.8376	0.6824	0.6858	0.081
et	Extra Trees Classifier	0.8357	0.9171	0.8001	0.8651	0.8290	0.6716	0.6763	0.248
svm	SVM - Linear Kernel	0.8184	0.0000	0.8387	0.8202	0.8257	0.6367	0.6430	0.011
dt	Decision Tree Classifier	0.8007	0.8004	0.8107	0.8023	0.8039	0.6010	0.6052	0.011
dummy	Dummy Classifier	0.5035	0.5000	1.0000	0.5035	0.6698	0.0000	0.0000	0.013

4.3. Shapley Feature Analysis

For better assessment, SHAP (Shapley Additive Explanations) is used to explain the most important features by visualizing the output (Figure 5). The first, second and third PCs are considered as component 1, component 2 and component 3, respectively. In this figure, feature values have two colors. Pink is used for features cause increasing and blue is used for showing the features which causes decreasing in prediction. As can be seen in this figure, first PC has the most impressive effect with the longest pink line, and the third PC has the lowest contribution.

4.4. Feature based RNN results

To avoid the disturbance caused by the gradient vectors, long short-term memory (LSTM) activation function has been used in connection with the active nodes in the RNN network.

In this situation, according to the input data consist of uncorrected visual acuity, degree of sphere, astigmatism, orientation of astigmatism and best corrected visual acuity, the LSTM activation function adapt the input and by considering the transient state value, it scales the output of the activation function. To determine the optimal value of weight coefficient, stochastic gradient descent method has been used. This proposed approach

has 6 layers. The first layer is the input layer and the second and third layers are used as LSTM layers. In the fourth layer, two actions are performed. In the first stage, the output of the third layer is connected with the input vector, and in the next stage, a linear combination of the inputs is produced in each time step. The fifth and sixth layers include a multilayer perceptron neural network. A hidden layer is embedded in this network. The complete specifications of the proposed method are listed in Table 5.

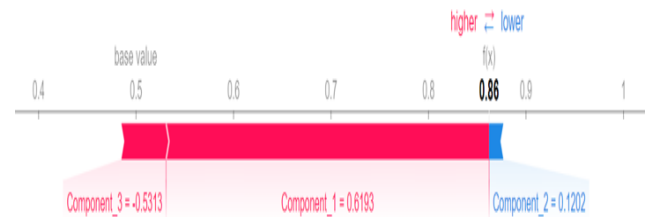


Figure 5: SHAP single observation

The output of the neural networks are the five characteristics of eye vision. The RNN used in this research is a neural network with 44 hidden neurons in hidden layer. From the available data, 85% were used for training and the rest for evaluation and validation of each of the trained networks. By comparing the coefficient of cell size changes in the two groups of keratoconus patients and the group of healthy individuals, no significance difference was found ($P = 0.828$). By examining the specular microscopy images, changing the shape of the cells, people with keratoconus only in seven eyes (9.26%) and in the control group, it was seen only in six eyes (24%). According to the amount of ratio equal to 1.1, the chance of cell shape change in the corneal endothelium of patients with keratoconus was 10% higher than that of healthy individuals. According to the confidence interval of 0.95 which was calculated between 0.59 - 1.095, it can be said that the cell shape change in the corneal endothelium of these two cases and control groups was similar to each other. Since by comparing the average cell density in two groups of keratoconus patients and healthy individuals (control group), no significant statistical difference was seen as equal to $P = 0.96$. The cell density in the upper part of the cornea is the highest in the two groups, but the interaction of the group (being sick or healthy) on the cell density was not significant ($P = 0.96$) (Table 5).

Table 5: Characteristics of the RNN method

No. of images	900	Type of network	RNN + LSTM
Hidden neurons	44	Function of hidden layer	Log - Sigmoid

Coefficient optimizer	Stochastic gradient descent	Function of output layer	Tan - Sigmoid
-----------------------	-----------------------------	--------------------------	---------------

Table 6: Comparison of the average difference in cell density of different areas of the cornea

Region 1	Region 2	Mean of difference (Region 1- Region 2)	Sig.
Central	Superior	-335.3	0.000
	Inferior	-28.9	0.994
	Nasal	-24.1	0.997
	Temporal	-107.5	0.529
Superior	Central	335.3	0.000
	Inferior	306.4	0.000
	Nasal	311.2	0.000
	Temporal	227.8	0.010
Inferior	Central	289	0.994
	Superior	-306.4	0.000
	Nasal	4.8	1.000
	Temporal	-78.6	0.787
Nasal	Central	24.1	0.997
	Superior	-311.2	0.000
	Inferior	-4.8	1.000
	Temporal	-83.4	0.748
Temporal	Central	107.5	0.529
	Superior	-227.8	0.010
	Inferior	78.6	0.787
	Nasal	83.4	0.748

4.4.1. Scenario 1: 6 months after Keraring operation

In predicting the visual characteristics of eyes with keratoconus six months after keraring implantation with the help of RNN, the regression result of training data is obtained as equal to $R_{\text{Training}} = 0.99997$. It implies a very close proximity between output values and network objectives in the training set (Figure 6). In the graph labeled Test, the regression analysis of the evaluation data set is presented. For evaluation data and for all data, the regression value is equal to $R_{\text{Test}} = 0.94889$ and $R_{\text{All}} = 0.98615$, respectively.

4.4.2. Scenario 2: 12 months after Keraring operation

For predicting the visual characteristics of the keratoconus after twelve months of keraring ring implantation, special RNN structure is designed (Figure 7). The regression of training data is $R_{\text{Training}} = 0.99992$, which indicates a very close proximity of the network's outputs to goals in the training set. In the evaluation level, the regression value is obtained as $R_{\text{Test}} = 0.92344$ and for all data, the regression value is calculated as $R_{\text{All}} = 0.98165$.

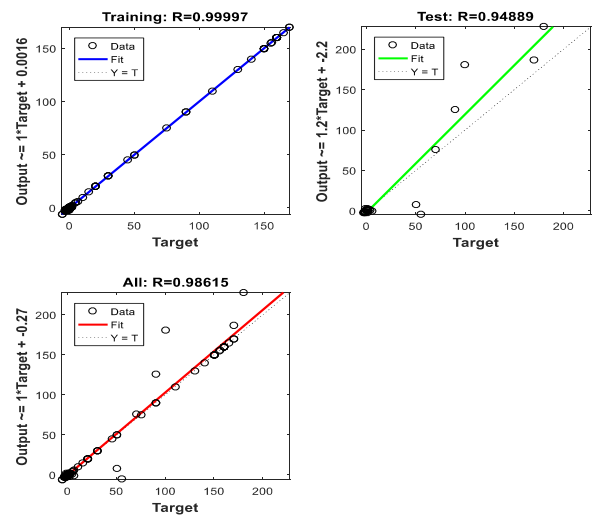


Figure 6: Results of predictive analysis of visual characteristics (Scenario 1)

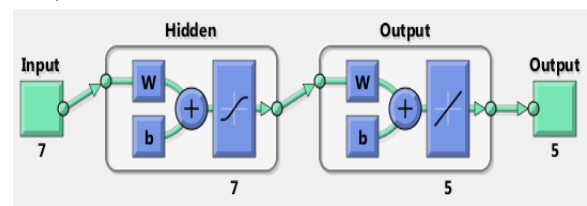


Figure 7: RNN structure for scenario 2

4.4.3. Scenario 3: 6 months after Myoring

In predicting As can be seen in Figure 8, in predicting the visual characteristics of eyes with keratoconus six months after myoring ring implantation using the RNN, the regression for training data is $R_{\text{Training}} = 0.99881$. Also, in the graph of the relationship between the output and the target, the fitted line has an angle close to 45° which indicates the high similarity between the outputs and the targets in the training set. The regression value for evaluation data is $R_{\text{Test}} = 0.92021$ and for all data $R_{\text{All}} = 0.98237$ which is close to unit (perfect match).

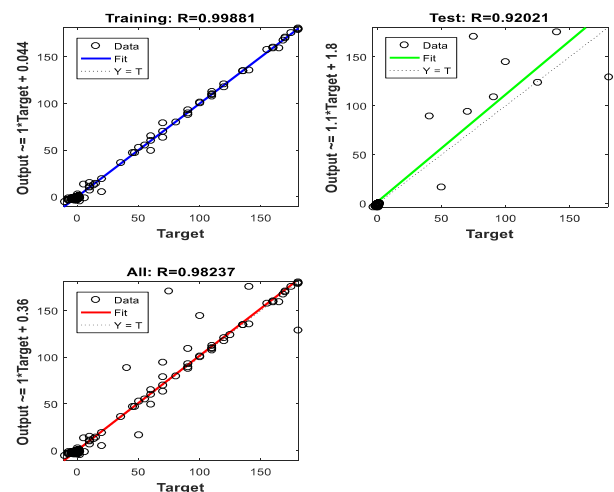


Figure 8: Results of predictive analysis of visual characteristics (6 months after Myoring)

4.4.4. Scenario 4: 12 months after Myring

For predicting the visual characteristics of eyes with keratoconus complications twelve months after myring ring implantation, special RNN structure has been used (Figure 9). The training data set, has a regression of $R_{\text{Training}} = 0.99968$ and for the evaluation data, the regression value is $R_{\text{Test}} = 0.90891$. From the regression analysis for all data (training and evaluation), $R_{\text{All}} = 0.95364$ was obtained.

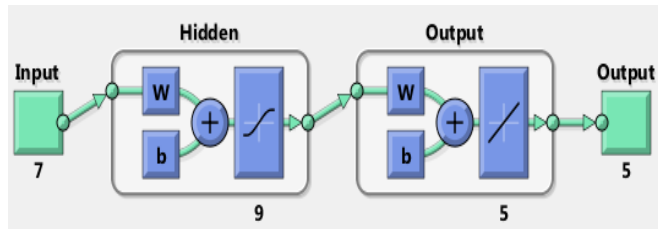


Figure 9: RNN structure for scenario 4

4.5. Results Comparison

In this section, the results of the proposed method are compared with KerNet, LeNet and AlexNet. In the first stage, the training accuracy versus number of features for different methods are compared with each other and the results obtained are shown in Figure 9. As can be seen, as the number of features increases, the accuracy of the obtained results decreases for all methods. At the same time, the proposed approach brought the least amount of accuracy reduction and this indicates that by increasing the number of features, using the proposed approach provides valuable results.

By selecting 100 features, the accuracy of the proposed approach is more than 94%. This dedicates that the designed approach has less sensitivity to the increase in the number of features than other traditional methods and so, the proposed approach can easily be used for problems on larger scales. In the following, criteria including Precision, Recall, F1-score and ACC have been used to compare the methods. The results of comparison are shown in Table 6.

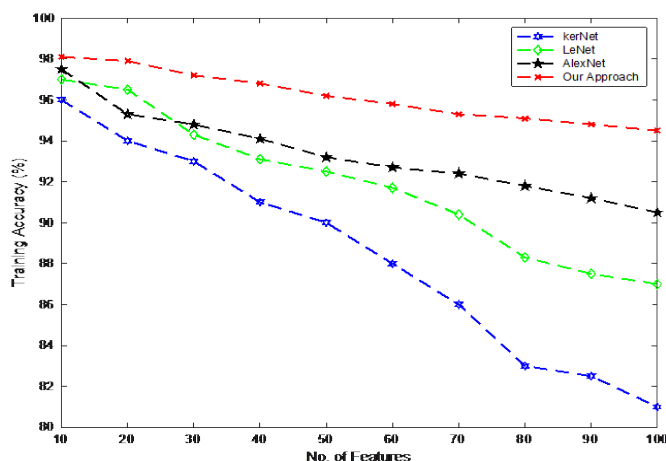


Figure 9: Comparison training accuracy for different methods

Table 7: Comparison of different metrics between methods

Model	Evaluation Metrics (%)			
	ACC	Recall	Precision	F1-Score
Our Approach	94.83	94.15	95.07	94.14
KerNet	94.74	93.71	94.10	93.89
LeNet	83.63	81.02	82.27	80.94
AlexNet	92.40	90.66	91.64	91.08

5. Conclusion

Keratoconus disease in mild and moderate stages has an effect on cell density, change in cell shape and endothelium. The most advanced available treatment for this disease is intracorneal ring implantation surgery. This surgery is used in very advanced cases of the disease or in some mild to moderate hunchbacks where it is not possible to use contact lenses due to the shape of the cornea. Considering that the success rate of the surgery in keratoconus patients are different and if the treatment is unsuccessful, the patient will need a corneal transplant. One of the concerns of ophthalmologists is which of the patients is a good candidate for this surgery and for whom this method brings better visual results. For this purpose, 900 images of Cornea were used and 7 morphological features were defined. These features consist of Area, Major Axis Length, Minor Axis Length, Convex Area, Perimeter, Eccentricity & Extent. For feature extraction Principal Component Analysis as well as three classification methods including logistics regression (LR), k-nearest neighbors (KNN) and decision tree (DT) algorithms were used. At the principal component analysis (PCA) stage, the contribution of different PCs were determined. Based on it's results, the first PC had a contribution of more than 69%, the variance of the first two PCs was about 89%. Hence, the feature dataset could be reduced from 7 to 2.

For comparison the results of different methods, 810 samples were used for determination of models' parameters and the rest of them were preserved for prediction accuracy analysis. In the next part, the obtained results from applying different classification methods such as LR, KNN and DT on original dataset were compared to each other. By the results, the logistic regression (LR) had the best performance with 88.52% accuracy. After applying PCA method based on first two PC, the best accuracy belonged to KNN method (more than 86%). In the last part of this paper, SHAP (Shapley Additive Explanations) was used to more explain the most important features by visualizing the output. the corneal topography is used as an input to a recurrent neural network (RNN) to identify whether or not the patient has keratoconus. By refining the settings of the RNN, the test set accuracy of the proposed technique was enhanced to 99.33%. Based on the obtained results, the

average prediction error of the visual characteristics of a patient with keratoconus six and twelve months after the Kraring ring implantation using RNNs with eight and seven neurons in the hidden layer is calculated as 9.82% and 9.29%, respectively. In order to predict the visual characteristics of a patient six and twelve months after Myoring ring implantation, RNNs with five and nine neurons in the hidden layer were used, and the average error of estimating characteristics was calculated as 11.46%, and 7.47%, respectively.

Conflict of Interest

The authors declare no conflict of interest.

References

- [1] J. Santodomingo-Rubido, G. Carracedo, A. Suzaki, C. Villa-Collar, S. J. Vincent, "Keratoconus: An updated review," *Contact Lens and Anterior Eye*, vol. 45, no. 3, 101559, 2022, doi.org/10.1016/j.clae.2021.101559.
- [2] A. Lavric, and P. Valentin, "KeratoDetect: keratoconus detection algorithm using convolutional neural networks," *Computational intelligence and neuroscience*, 2019, doi.org/10.1155/2019/8162567.
- [3] M. M. Vandevenne et al., "Artificial intelligence for detecting keratoconus," *Cochrane Database of Systematic Reviews* 11, 2023, doi.org/10.1002/14651858.CD014911.pub2.
- [4] V. Galvis, T. Sherwin, A. Tello, J. Merayo, R. Barrera, and A. Acera, "Keratoconus: an inflammatory disorder?," *Eye* 29, no.7, 2015, 843-859, doi.org/10.1038/eye.2015.63.
- [5] M.J. Kaisania, "A machine learning approach for keratoconus detection." (Ph. D Thesis, 2021).
- [6] P. J. Shih, H. J. Shih, I. J. Wang, and S. W. Chang. "The extraction and application of antisymmetric characteristics of the cornea during air-puff perturbations," *Computers in Biology and Medicine*, no. 168, 107804, 2024, doi.org/10.1016/j.compbiomed.2023.107804.
- [7] M. F. Greenwald, B. A. Scruggs, J. M. Visliser, and M. A. Greiner. "Corneal imaging: an introduction." *Iowa City (Iowa): Department of Ophthalmology and Visual Sciences, University of Iowa Health Care* 9, 2016.
- [8] A., H. Alyaa, H. N. Ghaeb, and Z. M. Musa. "Support vector machine for keratoconus detection by using topographic maps with the help of image processing techniques." *IOSR Journal of Pharmacy and Biological Sciences*, vol. 12, no. 6, 2017, 50-58, doi.org/10.9790/3008-1206065058.
- [9] M. C. Arbelaez, F. Versaci, G. Vestri, P. Barboni, and G. Savini. "Use of a support vector machine for keratoconus and subclinical keratoconus detection by topographic and tomographic data." *Ophthalmology* vol. 119, no. 11, 2012, 2231-2238, doi.org/10.1016/j.ophtha.2012.06.005.
- [10] A. Lavric, P. Valentin, T. Hidenori, and S. Yousefi. "Detecting keratoconus from corneal imaging data using machine learning." *IEEE Access* vol. 8, 2020, 149113-149121, doi.org/10.1109/ACCESS.2020.3016060.
- [11] K. Kamiya, Y. Ayatsuka, Y. Kato, F. Fujimura, M. Takahashi, N. Shoji, Y. Mori, and K. Miyata. "Keratoconus detection using deep learning of colour-coded maps with anterior segment optical coherence tomography: a diagnostic accuracy study." *BMJ open* vol. 9, no. 9, 2019, doi.org/10.1136/bmjopen-2019-031313.
- [12] L. M. Hartmann et al., "Keratoconus Progression Determined at the First Visit: A Deep Learning Approach With Fusion of Imaging and Numerical Clinical Data," *Translational Vision Science & Technology*, vol. 13, no. 5, 2024, doi.org/10.1167/tvst.13.5.7
- [13] A. Tillmann et al., "Acute corneal melt and perforation - a possible complication after riboflavin/UV-A crosslinking (CXL) in keratoconus.," *American journal of ophthalmology case reports*, vol. 28, 101705, 2022, doi.org/10.1016/j.ajoc.2022.101705
- [14] A. H. Al-Timemy, N. H. Ghaeb, Z. M. Mosa, and J. Escudero. "Deep transfer learning for improved detection of keratoconus using corneal topographic maps." *Cognitive Computation* vol.14, no. 5, 2022, 1627-1642. doi.org/10.1007/s12559-021-09880-3
- [15] K. Kazutaka, Y. Ayatsuka, Y. Kato, N. Shoji, Y. Mori, and K. Miyata. "Diagnosability of keratoconus using deep learning with Placido disk-based corneal topography." *Frontiers in Medicine* vol. 8, 2021, 724902, doi.org/10.3389/fmed.2021.724902
- [16] I. Issarti, A. Consejo, M. Jiménez-García, S. Hershko, C. Koppen, and J. J. Rozema. "Computer aided diagnosis for suspect keratoconus detection." *Computers in biology and medicine* vol. 109, 2019, 33-42. doi.org/10.1016/j.compbiomed.2019.04.024
- [17] B. R. Salem, and V. I. Solodovnikov. "Decision support system for an early-stage keratoconus diagnosis. " *In Journal of Physics: Conference Series*, vol. 1419, no. 1, p. 012023. IOP Publishing, 2019, doi.org/10.1088/1742-6596/1419/1/012023
- [18] X. Xu, T. Liang, J. Zhu, D. Zheng, and T. Sun. "Review of classical dimensionality reduction and sample selection methods for large-scale data processing," *Neurocomputing*, vol. 328, 2019, 5-15, doi.org/10.1016/j.neucom.2018.02.100
- [19] H. S. Hippert, C. E. Pedreira, and R. C. Souza. "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Transactions on power systems* vol. 16, no. 1, 2001, 44-55, doi.org/10.1109/59.910780
- [20] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling. "Deep learning for estimating building energy consumption. "Sustainable Energy, Grids and Networks 6, 2016, 91-99, doi.org/10.1016/j.segan.2016.02.005
- [21] M. R. Arahal, A. Cepeda, and E. F. Camacho. "Input variable selection for forecasting models. " *IFAC Proceedings*, Vol. 35, no. 1, 2002, 463-468, doi.org/10.3182/20020721-6-ES-1901.00730
- [22] Fan, Cheng, Fu Xiao, and Yang Zhao. "A short-term building cooling load prediction method using deep learning algorithms. " *Applied energy* vol. 195, 2017, 222-233, doi.org/10.1016/j.apenergy.2017.03.064
- [23] S. Hochreiter, and J. Schmidhuber. "Long short-term memory. " *Neural computation* vol. 9, no. 8, 1997, 1735-1780, doi.org/10.1162/neco.1997.9.8.1735

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).

A Computational Approach for Recognizing Text in Digital and Natural Frames

Mithun Dutta¹ , Dhonita Tripura¹, Jugal Krishna Das²

¹Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati-4500, Bangladesh

²Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka-1342, Bangladesh

*Corresponding author: Mithun Dutta, mithundutta92@gmail.com

ABSTRACT: Acquiring tenable text detection and recognition outcomes for natural scene images as well as for digital frames is very challenging emulating task. This research approaches a method of text identification for the English language which has advanced significantly, there are particular difficulties when applying these methods to languages such as Bengali because of variations in script, morphology. Text identification and recognition is accomplished on multifarious distinct steps. Firstly, a photo is taken with the help of a device and then, Connected Component Analysis (CCA) and Conditional Random Field (CRF) model are introduced for localization of text components. Secondly, a merged model (region-based Convolutional Neural Network (Mask-R-CNN) and Feature Pyramid Network (FPN)) are used to detect and classify text from images into computerized form. Further, we introduce a combined method of Convolutional Recurrent Neural Network (CRNN), Connectionist Temporal Classification (CTC) with K-Nearest Neighbors (KNN) Algorithm for extracting text from images/ frames. As the goal of this research is to detect and recognize the text using a machine learning-based model a new Fast Iterative Nearest Neighbor (Fast INN) algorithm is now proposed based on patterns and shapes of text components. Our research focuses on a bilingual issue (Bengali and English) as well as it producing satisfactory image experimental outcome with better accuracy and it gives around 98% accuracy for our proposed text recognition methods which is better than the previous studies.

KEYWORDS: Iterative, Component, Recognition, Bilingual

1. Introduction

Recently, a notable surge in research efforts dedicated to text recognition, a critical aspect within various image processing and vision algorithms. Scene text recognition, in particular, emerges as a challenging yet highly advantageous endeavor involving the identification of text within natural images. Deciphering text embedded in digital photographs poses a significant challenge. Moreover, image text recognition not only forms the foundation of information retrieval but also plays a crucial role in enabling effective human-machine communication. The potential applications of a system capable of identifying and extracting text from real-world images are extensive. Additionally, databases comprise texts of diverse types, including manually edited caption texts and scene texts with various orientations, thereby introducing layers of complexity to text detection and recognition. Notably, viewers often prioritize text when interpreting

images, emphasizing the significance of text detection and recognition in aiding human comprehension of intricate visual compositions.

Numerous methodologies have been developed to address the challenges associated with text detection and recognition in digital frames and natural scene images characterized by varying orientations, scripts, font sizes, and other factors. Previous studies suggest that many existing methods focus on specific data types and address particular issues such as complex backgrounds, low contrast, or multiple scripts and orientations, resulting in suboptimal performance when confronted with data influenced by multiple adverse factors [1, 2, 3]. The primary challenges stem from various sources: frames captured with low-resolution cameras which make low contrast inputs; images also taken with high-resolution devices may exhibit high contrast input but frequently feature complex backgrounds, leading to an increased incidence of false positives; images sourced

from digital platforms often showcase a multitude of character components.

In this study, we employed mask-RCNN for character and text detection and recognition of both Bangla and English sentences. R-CNN (Region-based Convolutional Neural Network), including its variant Mask R-CNN, represents a class of machine learning models tailored explicitly for computer vision tasks, particularly object detection. Mask R-CNN propagates Faster R-CNN by introducing an additional pulse for outputting object masks alongside class labels and bounding-box offsets, thereby facilitating the extraction of finer spatial layouts of objects. Additionally, our proposed method incorporates the Fast Iterative Nearest Neighbor (Fast INN) algorithm, which utilizes shape information to detect candidate components. Subsequently, Fast INN extracts shape of script. The novelty of our approach is to explore these fundamental concepts to address the ongoing challenges of text detection and recognition without imposing rigid constraints, thereby aiming for enhanced performance and versatility.

2. Literature Review

In [1], the authors proposed a method for detecting and tracking text on a variety of large and small text blocks. This consists of two different modules: first one is a sum of squared difference (SSD) and another one is a contour-based module. The main purpose of the paper proposed [2] text recognition, which submitted a petition on the text lines detection.

In [3, 4], the authors proposed the text detection method which is carried out through edge detection, local thresholding, and hysteresis edge recovery. In [5, 6], the researches proposed an SVM classifier which used to identify text from the selected features object. In [7, 8], the authors addressed a model that is based on invariant features. In [9, 10], the authors proposed signal extraction techniques based on RWT method for extracting text information. They also proposed text line extraction based on integrated K-shortest path optimization. In [11, 12], the authors used the NLP to analyze and become acquainted the data set and also used both Naive Bayes and logistic regression algorithms to determine the best accuracy system.

3. Proposed System

3.1. Proposed Framework

In this section, we outline the operational procedures of the proposed machine learning-based text recognition framework and delve into its logical execution. The experiment integrates a preprocessing technique tailored to mitigate unwanted noise and artifacts effectively. Initially, we employ the Connected Component Analysis (CCA) model to determine adjacent pixels based on

predefined pixel connectivity criteria. Additionally, we utilize edge detection methodologies. Typically, false values in the image are associated with background pixels, while valid values represent foreground or object pixels. The connected component analysis methodology segments the image into regions, identifies relevant areas, and extracts text from these regions. Subsequently, the image is partitioned into smaller components, which are then classified based on their geometric attributes.

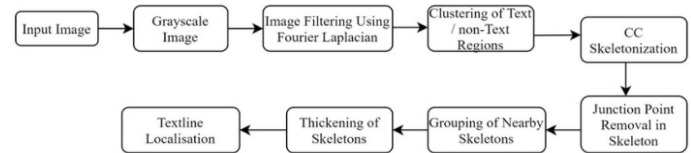


Figure 1: Text localization based on connected component analysis

Text localization involves the utilization of conditional random field (CRF), a statistical modeling technique employed when class labels for different inputs which are independent. Specifically, in image segmentation, the assignment of a class label to a pixel is contingent upon the labels of its neighboring pixels. To calculate the confidence of region texts, translate the output, conditioned variable (s), $s \in \{\text{acc}, \text{rej}\}$, and t can be calculated based on the Bayes' theorem, where $P_t(s|x)$ are calculated is as follows.

$$P_t(x|s) = \frac{P_t(s|x)P_t(x)}{\sum_x P_t(s|x)P_t(x)} = \frac{P_t(s|x)P_{t-1}(x|\text{accept})}{\sum_x P_t(s|x)P_{t-1}(x|\text{accept})}$$

Image Segmentation: Niblack's binarization formula [13] is defined as:

$$b(x) = \begin{cases} 0 & , \text{ if } \text{gray}(x) < \mu_r(x) - k \cdot \sigma_r(x); \\ 255 & , \text{ if } \text{gray}(x) > \mu_r(x) + k \cdot \sigma_r(x); \\ 100 & , \text{ other,} \end{cases}$$

And an implementation, $P(Y|X)$ is as:

$$P(Y|X) = \frac{1}{Z(X)} \exp(-E(X, Y, N, \lambda)),$$

Where $Z(X)$ is the normalization constant which is used to solve the problem, and the best label Y^* might be considered by maximizing conditional probability $P(Y|X)$ to minimize the energy.

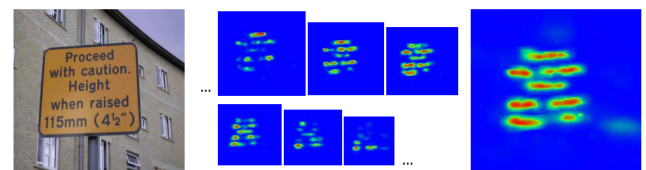


Figure 2: (a) original image (b) text confidence maps (c) text confidence map for the original image

Then, Mask R-CNN is an advance computer vision model used for object instance segmentation. A text detector based on Mask R-CNN is used, and fully convolutional

networks mainly inspire the methods. First, CNN is adopted to detect text blocks from which character candidates are extracted.

Then, FPN (Feature Pyramid Network) is used to predict the corresponding segmentation masks. Convolutional Neural Network (CNN) is a one kind of AI based neural network which uses for recognizing and processing image components that is optimized to process pixel data. It integrates data components detection task where the goal is to detect object through the formation of bounding box prediction of an image and a semantic segmentation task, that may classify every pixel into pre-defined object categories.

This is an execution of Mask R-CNN on Python 3, and the model generates bounding boxes and segmentation masks for each data object in the frame, which is also based on FPN.

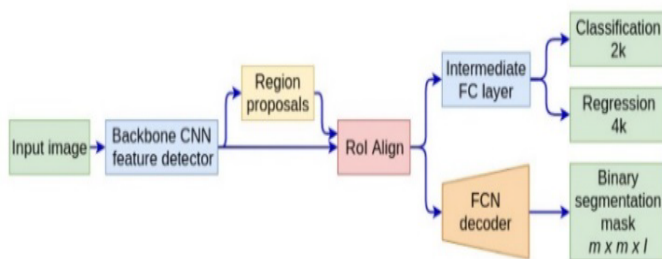


Figure 3: Flowchart of segmentation with Mask R-CNN

Mask R-CNN combines object detection and instance segmentation by using a Feature Pyramid Network (FPN) and the Region of Interest Align (ROIALign) layer, Mask R-CNN which achieves strong performance and accuracy rate.

Now, a combined method of Convolutional Recurrent Neural Network (CRNN), Connectionist Temporal Classification (CTC) with K-Nearest Neighbors (KNN) Algorithm for extracting text from images. Extracting text of different shapes and sizes, various directions and orientations from images, especially from web pages and sites which is augmented reality assistance systems, and content moderation in social media platforms with the combined method. Combining Convolutional Recurrent Neural Networks (CRNN) with Connectionist Temporal Classification (CTC) allows CRNN to handle variable-length sequences without requiring an explicit alignment between input images and text outputs.

Suppose a given sequence $X = [x_1, x_2, x_3, x_4, x_5, \dots, x_T]$, such as any voice, and by mapping the sequence to its corresponding outcome sequence $Y = [y_1, y_2, y_3, y_4, y_5, \dots, y_U]$. Our main object is to find out an accurate mapping between X 's and Y 's.

In particular:

- X and Y are different in length or dimension.
- The lengths ratio of X and Y may vary from each to other.
- No accurate alignment of X and Y .

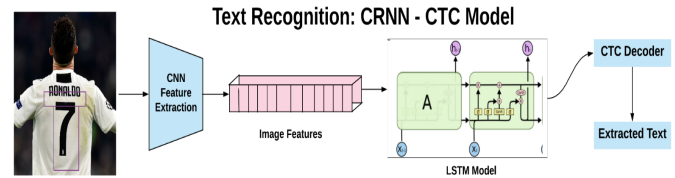


Figure 4: Text Recognition Pipeline Architecture

In CTC, a conditional probability $p(\pi|x)$ can be labeled as follows:

$$p(\pi|x) = \prod_{t=1}^T y_{\pi_t}^t, \forall \pi \in L'^T$$

$$p(s|x) = \sum_{\pi \in B^{-1}(s)} p(\pi|x)$$

CTC loss function is described as

$$L_{CTC} = -\ln p(s|x)$$

Further, the k-nearest neighbor (KNN) classifier uses proximity to make classifications or predictions about the grouping of an individual data point and fundamentally relies on a distance metric. The most common choice is the Minkowski distance $\text{dist}(x,z) = (\sum_{r=1}^d |x_r - z_r|^p)^{1/p}$. Finally, the proposed method, Fast Iterative Nearest Neighbor algorithm (Fast INN), is applied to recognize the text. The FINN algorithm is suggested by the inspiration of KNN, which is one of the simplest Machine Learning algorithms. The proposed algorithm estimates the similarity between the new case/data and the available cases and places the new case in the category that is most similar to the available data. During the training phase, the FINN algorithm stores the datasets, and when it gets new data, it then classifies that data into a category that is similar to the latest data. As the FINN algorithm helps to identify the nearest points or the groups for a query point and to determine the closest groups or the nearest points for a query point, we need to calculate some distance metrics (Euclidean distance, Manhattan distance, Minkowski distance, etc.).

Our Fast Iterative Nearest Neighbor (Fast I-NN) working can be explained through the following algorithm:

Step-1: Select the number of the neighbors (Suppose the number is N)

Step-2: Calculate the distance (Euclidean) of N number of neighbors

Step-3: Take the N nearest neighbors as per the calculated Euclidean distance.

Step-4: Among these N neighbors (adjacent data points), count the number of the data points in each category.

Step-5: Assign the neoteric data points to that category

Step-6: Find for which the number of the neighbor is maximum.

Step-7: Calculate the distance is minimum.

Step-8: Compare the new data pixel with residual data points to that category.

Step-9: Predict and Print the output.

3.2. Datasets

In this study, research will use most of the original dataset images, which are taken with a Kodak DX7590 (5.0 MP) still frame camera and a Sony DCR-SR85E handy camera in still mode, as well as from Kaggle. From different background images are taken like roadways, digital banner, signboard, wall writings, etc. Data may be prepared for a machine learning approach using Python language through Jupyter Notebook. The proposed text recognition method is admired in both cases qualitatively and quantitatively. Datasets include original images of dimension either (321x481) or (481x321) pixels, and from the datasets randomly, 80 percent of images are used for training and 20 percent of images for testing. For Augmentation, we used to tilt the image with different angle, zooming, rotation, shearing width and height shifting, and horizontal and vertical flipping techniques in a Python setup environment.

3.3. Configure and Train the Model

The model, Default-Trainer, Default-Predictor, Color-Mode, and Visualizer were all imported. We also imported mask-RCNN with FPN, as well as other models and the associated checkpoints. We defined the dataset and other settings, such as total number of workers, batch size, as well as the number of classes, by importing Default-Trainer. We trained the model further after initializing it using pre-trained weights. We've trained our new model in Jupyter Notebook and Google Colab wherein it could dash and make advantages. The system generated an output folder, into which the trainer saved

the training checkpoints to count the maximum and best checkpoints for the trained model. The trained software automatically saves the checkpoints in the output directory during training.

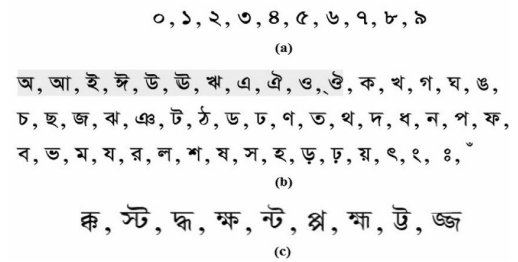


Figure 5: Bangla character (a) Digits (b) Vowels and consonants (c) Compound/joint characters



Figure 6: Frequently used grapheme roots of Bangla Char

3.4. Eliminate unwanted Character

This system eliminates multiple-identified characters from the object array after replacement. Machine learning detection models can predict the same object or predict the same instance with a different class at another time. Consequently, we must remove redundant objects from the array, keeping only one instance wherein the same instance is not recognized more than once.

3.5. Recognize the Text

All the images previous step is used for both to train and test [9, 10]. Our proposed method has been accustomed for text line recognition. The following figure shows the step of capturing to recognize.



Figure 7: From image capture to text recognition

4. Experimental Result

4.1. Performance Evaluation

Maximum accuracy rate indicates the perfect prediction for the localization and recognition of text. A

minimal false positive rate indicates fewer redundant pixels in unwanted regions resulting in a higher accuracy rate. This provides a balanced measure of the model's ability to correctly identify both positive and negative instances. In this paper, we have adapted the performance of machine learning models. The figure below reincarnates the results of the trained machine learning. Here, Figure a. shows accuracy rate of performance, and Fig b. shows false negative prediction, Fig c. indicates accuracy of the foreground classification, Fig d. shows the accuracy rate of mask RCNN, Fig e. represents the false negative of mask RCNN, and lastly, Fig f. shows the total loss of the mask RCNN.

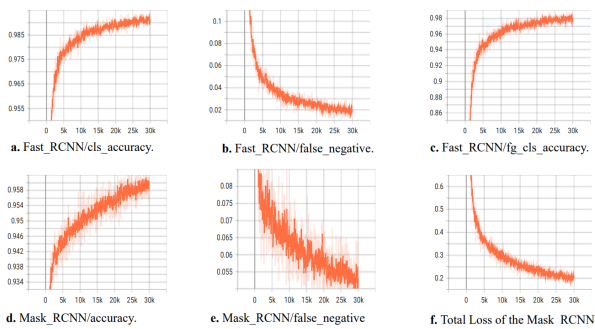


Figure 8: Result of the Trained Machine Learning Model

4.2. Experimental Result and Discussion

Our proposed model for future bilingualism text recognition will perform very well. We have used a huge number of data set to acquire a better performance and a well predicted result. To achieve the performance and print the result we had to perform our operational work properly in several sectors from where quite a few data were lost. Using the proposed algorithm through python language significant percentage accuracy rate is achieved. To calculate the sample size and accuracy rate we need the total number of successfully recognized images. The following table depicts the results of our study of all steps together.

Table 1: Summarizes the accuracy in terms of its gradations

Gradations	Text Detection	Text Extraction	Text Recognition
Total Data Samples	19200	19200	19200
Correct for Bangla	18430	18110	17998
Accuracy for Bangla	95.98%	94.32%	93.74%
Correct for English	19110	19030	18810
Accuracy for English	99.53%	99.11%	97.97%

5. Conclusion

In this research, we have well-acquainted and analyzed various text detection techniques for natural scenes as well as for digital images. In here, we introduced a unique model for detecting and recognizing the Bangla and English text using machine learning and the proposed Fast Iterative Nearest Neighbor (FINN) method. Before the system detects text, images are initially taken from the nature or surrounding environment and from that image texts are extracted. After that, text recognition was done by a merge proposed method where it gave false positive results or was unable to recognize some words or parts of them [13]. Then, the proposed method works effectively on curved or slanted Bengali and English texts as well. Pre-processing techniques make this model more robust against any random artifacts or unwanted blurring effects. However, the system gives about 98% accuracy for the proposed text recognition model.

References

- [1] H. Li, D. Doermann, and O. Kia, "Automatic Text Detection and Tracking in Digital Video," *IEEE Transactions on Image Processing*, vol. 9, no. 1, 147-156, Jan. 2000, doi: 10.1109/83.817607.
- [2] D. Chen, J.-M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," *Pattern Recognition*, vol. 37, 595-608, 2004, doi:10.1016/j.patcog.2003.06.001.
- [3] P. Shivakumara, T. Q. Phan, and C. L. Tan, "A Laplacian Approach to Multi-Oriented Text Detection in Video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, 412-420, Feb. 2011, doi:10.1109/TPAMI.2010.166.
- [4] M. R. Lyu, J. Song, and M. Cai, "A Comprehensive Method for Multilingual Video Text Detection, Localization, and Extraction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, 243-255, Feb. 2005, doi:10.1109/TCSVT.2004.841653.
- [5] X.-C. Yin, Z.-Y. Zuo, S. Tian, and C.-L. Liu, "Text Detection, Tracking and Recognition in Video: A Comprehensive Survey," *IEEE Transactions on Image Processing*, 1-24, 2015, doi: 10.1109/TIP.2016.2554321.
- [6] Q. Ye, Q. Huang, W. Gao, and D. Zhao, "Fast and robust text detection in images and video frames," *Image and Vision Computing*, vol. 23, 565-576, 2005, doi:10.1016/j.imavis.2005.01.004.
- [7] K. S. Raghunandan, P. Shivakumara, S. Roy, G. H. Kumar, U. Pal, and T. Lu, "Multi-Script-Oriented Text Detection and Recognition in Video/Scene/Born Digital Images," *IEEE Transactions on Circuits and Systems for Video Technology*, doi: 10.1109/TCSVT.2018.2817642, 2018.
- [8] M. Cai, J. Song, and M. R. Lyu, "A New Approach for Video Text Detection," in *Proceedings. International Conference on Image Processing*, 2002, 117-120, doi: 10.1109/ICIP.2002.1037973.
- [9] C. S. Shin, K. I. Kim, M. H. Park, and H. J. Kim, "Support Vector Machine-Based Text Detection in Digital Video," in *Proceedings of IEEE ICIP*, 2000, 634-641.
- [10] H. Wang, S. Huang, and L. Jin, "Focus On Scene Text Using Deep Reinforcement Learning," in *Proceedings of the 24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, Aug. 2018, 3759-3766, doi: 10.1109/ICPR.2018.8545022.
- [11] Y. Wang, "Extraction Algorithm of English Text Information from

Color Images Based on Radial Wavelet Transform," *Special Section on Gigapixel Panoramic Video with Virtual Reality*, Aug. 2020, doi: 10.1109/ACCESS.2020.3020621.

- [12] O. Y. Ling, L. B. Theng, A. Chai, and C. McCarthy, "A Model for Automatic Recognition of Vertical Text in Natural Scene Images," in *2018 8th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia, 2018, doi: 10.1109/ICCSCE.2018.8685019.
- [13] M. Dutta, A. Mohajon, S. Dev, D. S. Bappi, and J. K. Das, "Text Recognition of Bangla and English Scripts in Natural Scene Images," *International Journal of Advanced Research in Science and Technology*, vol. 12, no. 10, 1137-1142, 2023.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).