Jordanian Journal of Computers and Information Technology

JJCIT

www.jjcit.org   jjcit@psut.edu.jo

An International Peer-Reviewed Scientific Journal
Financed by the Scientific Research Support Fund

JJCIT

Jordanian Journal of Computers and Information Technology (JJCIT)

The Jordanian Journal of Computers and Information Technology (JJCIT) is an international journal that publishes original, high-quality and cutting edge research papers on all aspects and technologies in ICT fields.

JJCIT is hosted by Princess Sumaya University for Technology (PSUT) and supported by the Scientific Research Support Fund in Jordan. Researchers have the right to read, print, distribute, search, download, copy or link to the full text of articles. JJCIT permits reproduction as long as the source is acknowledged.

AIMS AND SCOPE

The JJCIT aims to publish the most current developments in the form of original articles as well as review articles in all areas of Telecommunications, Computer Engineering and Information Technology and make them available to researchers worldwide. The JJCIT focuses on topics including, but not limited to: Computer Engineering & Communication Networks, Computer Science & Information Systems and Information Technology and Applications.

INDEXING

JJCIT is indexed in:

EDITORIAL BOARD SUPPORT TEAM

LANGUAGE EDITOR	EDITORIAL BOARD SECRETARY
Haydar Al-Momani	Eyad Al-Kouz

JJCIT ADDRESS

WEBSITE: www.jjcit.org
EMAIL: jjcit@psut.edu.jo
ADDRESS: Princess Sumaya University for Technology, Khalil Saket Street, Al-Jubaiha
B.O. BOX: 1438 Amman 11941 Jordan
TELEPHONE: +962-6-5359949
FAX: +962-6-7295534

JJCIT

60

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

A SURVEY ON WIRELESS SENSOR NETWORK-BASED IOT DESIGNS FOR GAS LEAKAGE DETECTION AND FIRE-FIGHTING APPLICATIONS

Haythem Bany Salameh[1], Mohammad Dhainat[2] and Elhadj Benkhelifa[3]

ABSTRACT

The rapid advancements in wireless sensor network (WSN) technology gave impetus for large-scale deployment of Internet-of-things (IoT) services and applications. One of the envisioned IoT applications is the use of wireless sensor nodes in gas-leakage monitoring and detection applications. Such IoT applications can provide better protection to fire fighters and provide safety and early-warning gas detection alarms within a timely manner for individuals, factories and institutions. In this article, we highlight the unique characteristics of WSNs, discuss the main WSN design requirements associated with gas leakage and monitoring applications, discuss main differences between data collection- and event detection-based WSN solutions and present a detailed overview of the works that have been accomplished on providing WSN solutions for gas leakage detection and monitoring.

KEYWORDS

1. INTRODUCTION

The recent development of sensor technology along with the huge advances of the wireless communication and networking technologies resulted in moving toward an expected large-scale deployment of Internet-of-thing (IoT)-based wireless sensor networks (WSNs) (e.g., [1]–[4]). Large number of applications are envisioned for such type of wireless networks, such as environmental monitoring, military/civilian surveillance and security, precision agriculture, industrial automation, manufacturing and inventory control, transport monitoring and control, smart cities, smart homes, smart grid, …etc. (e.g., [5]–[9]).

An IoT-based WSN includes a large number of inter-connected small-size, inexpensive and low-energy dissipating devices [10], [11]. There are two types of deployment of such devices: random deployment or pre-specified manual deployment. The sensor devices are capable of self-organizing themselves to create a multi-hop wireless network. One of the most attractive features of IoT-based WSNs is the possibility of on-the-fly deployment of the sensor devices being used for unattended operations with minimum maintenance and no pre-existing infrastructure [12], [13]. Each sensor device can sense specific physical conditions or collect some required information (e.g., detect Liquefied Petroleum Gas (LPG) leakage [14]–[16]), process the collected/sensed data and send the reported measurements to a named pre-specified control node. Data aggregation and/or compression may be performed by the sensor nodes such that the communication overhead and energy consumption are reduced [17], [18]. Such attractive features and characteristics of WSNs made it a suitable candidate to be used in fire-fighting and LPG-gas detection applications [19], [20], [21].

LPG is one of the necessities of daily life, as almost all residential locations and service areas have gas cylinders (i.e., houses, factories, restaurants, hotels). However, any leakage of gas may cause the occurrence of explosions and serious fires. Thus, it is important to identify such leakage and take the appropriate action in a timely manner [22]–[23]. For this purpose, a WSN can be pre-deployed to remotely monitor and detect gas leakage in a timely manner. Such information can decrease the occurrence of explosions and serious fires and can assist in the process of controlling the spread of the gas leakage. On another hand, WSN technology along with IoT capabilities can provide better

1.   H. Bany Salameh is with Department of Telecommunications Eng., Yarmouk Uni., Irbid, Jordan. Email: `haythem@yu.edu.jo`
2.   M. Dhainat is with Civil Defence Directorate, Amman, Jordan. Email: `mohammad.fozi@cdd.gov.jo`
3.   E. Benkhelifa is with the School of Computing and Digital Tech., Staffordshire Uni., UK. Email: `E.Benkhelifa@staffs.ac.uk`

protection and safety to fire fighters, where the sensor nodes can be deployed on-the-fly by the firefighters operating in the hazard zone (e.g., burning building). The sensor nodes can wirelessly send information about the current status (such as gas density) inside the burning/gas leaked closed area to the internet or a command center. This allows the fire fighters to take the appropriate action before entering the dangerous area [24], [25], [26]. We note here that several WSN systems have been designed for fire-fighting, gas monitoring and detection applications based on different wireless communication technologies. However, most of these designs did not fully exploit the capabilities of IoT systems and WSN technology and did not consider their unique functionalities and requirements. Very few works have proposed LPG leakage detection systems based on the IoT technology, but the proposed systems are stand-alone sensing systems that do not utilize the WSN technology. The developed communication protocols, hardware designs and processing mechanisms for WSNs cannot be directly implemented in IoT-based WSNs. Therefore, new designs and communication protocols are needed to fully utilized the IoT functionalities and capabilities in WSNs such that network performance is improved (i.e., delay, throughput, energy consumption, connectivity …etc.).

The main objective of this paper is to overview and analyze the main WSN implementations and designs that have been proposed for fire-fighting and gas-monitoring applications in the literature. Specifically, a number of WSN system implementations and IoT deployments are surveyed, in which their operation details are explained. Furthermore, we briefly highlight the design requirements and specifications for fire-fighting and gas detection and monitoring applications. The different deployment possibilities of WSNs/IoT for such specialized applications are also summarized. In addition, open problems and potential research directions are provided.

The rest of this paper is organized as follows. Section 2 provides an overview of WSNs and their key design/deployment issues and challenges. In Section 3, we summarize the design requirements for an appropriate WSN design for fire-fighting and LPG gas detection applications and highlight the differences between data-collection and event-detection WSN designs. Section 4 provides an extensive overview of the various WSN and IoT solutions designed for gas-monitoring and fire-fighting applications. Section 5 provides a detailed discussion of the surveyed systems and highlights potential research directions. Finally, conclusion remarks are provided in Section 6.
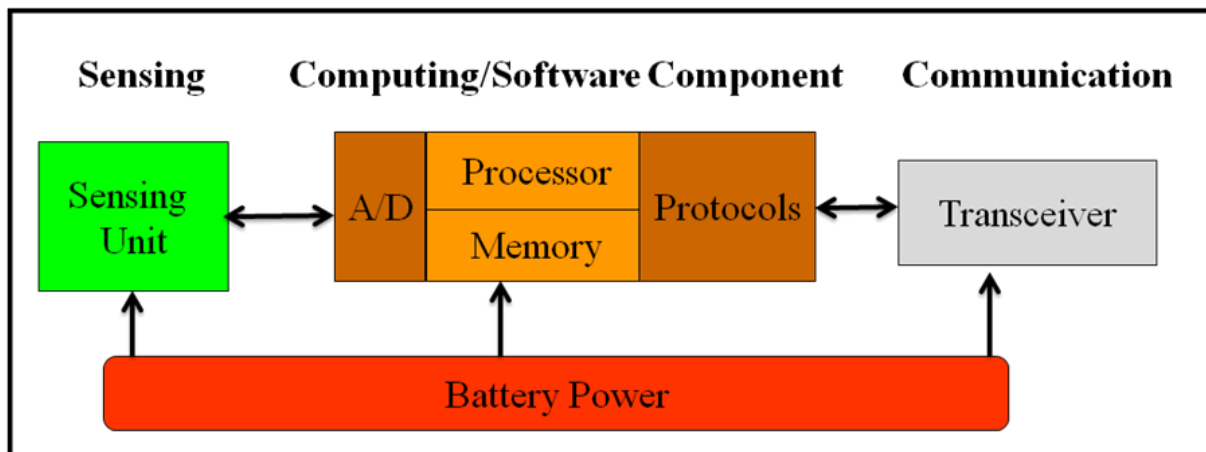
Figure 1. Sensor node architecture [24].

2. WSN ARCHITECTURE AND KEY DESIGN CHALLENGES

In this section, we present the overall architecture of WSNs, describe the functionality of the different elements of such architecture and discuss the main/common design challenges in providing efficient deployment and operation of WSNs such that a designer can take them into consideration when designing a WSN that is to be used for specified tasks.

2.1 WSN Architecture

A WSN is composed of a large number of small-size spatially distributed embedded sensor devices

62

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

with wireless networking capabilities that are responsible of monitoring and collecting specified physical information/events and reporting the sensed data to a centralized command device, referred to as a sink node. Sensor nodes are capable of self-organizing the communication between themselves and/or with the sink node through well-defined medium access control (MAC) and routing protocols, where single or multiple-hop communications are possible. The sensor nodes are usually battery-powered with extremely limited non-rechargeable energy sources. Therefore, sensor nodes perform data processing for gas-leakage and fire-fighting applications (e.g., aggregation, decision, fusion …etc.) of the collected raw information before conducting any transmission to save energy. WSNs measure environmental conditions, such as temperature, pollution level, gas-leakage occurrence, humidity level …etc. Several applications for WSNs with IoT capabilities are envisioned, including environment monitoring, health applications, militarily surveillance, industrial automation, smart cities, habitat research, search-and-rescue in hazard operating field and many other residential and industrial applications.

Several information and communication technologies are utilized in WSN systems including hardware components, software implementations and wireless networking capabilities. Specifically, a sensor node architecture is composed of 6 major inter-connected elements [24]: (1) A power-supply unit, which is, in general, a limited energy battery, (2) a transmission/reception unit, which is a radio frequency (RF) transceiver, (3) a sensing element, which depends on the physical condition that is to be monitored, (4) an analog-to-digital (A/D) converter unit, as most of the measured data are analog signals, (5) a micro-processor unit, which is needed to perform data aggregation and analysis and (6) a data storage unit, which is generally a memory with limited size. Figure 1 shows the architecture diagram of a sensor device.

2.2 Key Design Issues in WSNs

The key design challenges in providing effective WSN deployment and operation can be summarized as follows:

- Hierarchical architecture (clustering): Providing effective clustering mechanisms for WSNs is essential to provide a scalable approach to routing and network management in a large-scale WSN as well as prolonging network life-time.

- Coverage and redundancy elimination (using sleep/wakeup mode): Providing efficient deployment mechanisms that guarantee network connectivity and field coverage with minimum possible sensor redundancy is essential to prolonging network life-time without affecting coverage and connectivity.

- Channel access (MAC): Providing efficient MAC protocols that well-suit the unique characteristics of WSNs is important to guarantee proper functionality and preserve nodes' energy.

- Localization and attribute-routing: Providing efficient localization mechanisms is essential in the operation of WSNs, as most of the sensed data and measurements are attribute-based. Localization facilitates attribute-based routing, which indeed preserves network energy.

- Cooperative signal processing (multi-model sensing): Designing effective data aggregation and processing algorithms is very important to reduce communication overhead and prolong network life-time.

3. WSN DESIGN REQUIREMENTS AND OPERATION MODES FOR GAS MONITORING APPLICATIONS

3.1 Key Design Requirements

It is well-known that acquiring information from the inside of a burning/gas-leaked closed area is a challenging problem. Required information by the fire department/control unit can be remotely obtained using WSNs. The different information and the potential WSN solutions are given in Table 1 [27]. Depending on the application requirements, each sensor node can be equipped with one (or more) kind of the aforementioned sensitive sensor units.

Table 1. The different information needed for fire-fighting applications and sensing solutions.

Types of Needed Information	Needed Sensor Device
Information about the proximity of fire-fighters to danger.	Temperature, smoke, oxygen and olfactory sensors
Information required to decide on the likelihood of flashover.	Temperature sensor
Information required to decide on the possibility of backdrafts.	Oxygen sensor
Information needed to detest hidden.	Temperature and smoke sensors
Information needed to identify structural collapse issues.	Accelerometer sensor
Information regarding the existence of gas leaks (gas concentration).	Gas sensor

3.2 Operation Modes: Data-Collection *versus* Event-Detection Design

As stated before, two main types of WSNs can be implemented: pre-deployed and on-the-fly WSNs. The pre-deployed WSNs are categorized as event-detection networks, whereas an on-the-fly WSN can be characterized as a data-collection network. In fire-fighting applications, in which data collection and reporting are the main objectives, the sensors will be responsible of collecting data on short-duration basis and then sending information to a command centre. Thus, the sensor should remain awake during the operation. However, in the gas-leakage applications, where a WSN is to be used for event detection, the sensor nodes must stay into sleep mode if there is no event to report, which preserves their precious limited energy. Figure 2 depicts a network model for an IoT-enabled WSN that can be used for LPG monitoring.
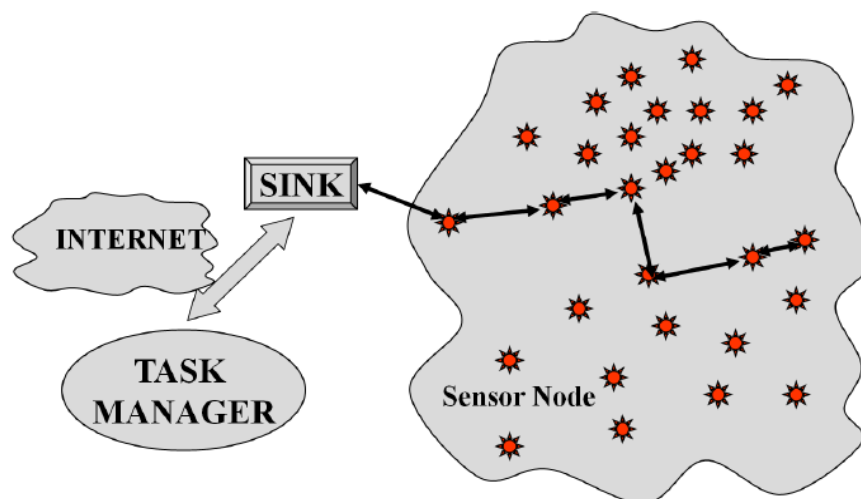
Figure 2. An illustrative WSN with a pre-deployed IoT-enabled WSN.

4. EXISTING WSN SOLUTIONS FOR GAS-MONITORING AND FIRE-FIGHTING APPLICATIONS

Several designs have been proposed to utilize the WSN technology fire-fighting and LPG-gas leakage monitoring and processing applications (e.g., [23]- [56]). In [23], [24], the authors proposed a multi-level cluster-based WSN for gas-leakage detection applications that provides early-warning alarms in case of LPG leakage. They provided communication protocols and hardware implementations based on cognitive radio technology. In [25], a wireless gas sensor network (WGSN) is designed for explosive gas detection. In case of gas leakage, the sink node sends an alert message to the command

64

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

center using GSM/GPRS or Ethernet connection. Such system can use a wireless actuator to autonomously control gas emission source.

In [26], the authors designed a gas-leakage sensing device with networking capability to form a WSN. The developed WSN creates a smart platform that collects, processes and analyzes the sensed information, in which an access to the sensed data is made available anywhere and anytime. A method for unauthorized access detection to oil or gas pipes is presented in [28]. This can be accomplished by measuring the protective cathodic voltage and reporting any differences in the measured voltage, which can be used as an indicator of a technical failure or an unauthorized access to the pipe. In [29], the authors proposed the use of the Electronics Line Ltd's wireless gas-leak detector. Such line is capable of detecting mixtures of air and hazard gases (e.g., butane, methane, natural gas and propane). In case of detecting gas leakage, the detector sends notification messages to the control panel. In [30], the authors proposed a practical gas-leak detection system that consists of two main modules: a detector with wireless capability module and a wireless hand-held remote module. The hand-held module is responsible of displaying the gas-leak level and rate as sensed by the detector in alphanumeric values. The remote module and the detector module communicate over a wireless channel to exchange data.

A WSN prototype for hydrogen gas-leak detection was manufactured to be used in hydrogen filling stations [31]. The manufactured WSN prototype consists of ten sensor devices that are able to detect hydrogen gas leakage and track the spatial distribution of the leaks. In [32], the author developed a WSN to provide safety in Petrochemical Industry. The developed WSN is meant to reduce the time of rescue operation. The system in [32] can wirelessly report the data from the monitoring sites to the command controller (sink). The authors in [32] proposed two design variants for their WSN system: fixed-point and dynamic-mobile monitoring variants. A WSN for monitoring gas emission levels in industrial facilities is developed in [33]. When the emission level increases above a threshold level, the proposed WSN generates alarm messages containing the emission levels.

In [34], the authors developed a context-aware WSN system that provides tacit networking between fire-fighters. Each fire-fighter is equipped with a WiFi capable PDA that contains a sensor device. The sensor device in the PDA collects the required information from other nodes that are pre-existing inside the building to notify the fire-fighters of any danger and possible hazards. The pre-allocated sensors are also used as location anchors, which enables the fire-fighters to explore their path inside the building. The paper in [35] addressed the high energy consumption of the LPG sensor unit, which can result in fully depleting nodes' energy keeping no enough energy to transmit alarm messages. To solve this problem, the authors advised the use of two power supplies: digital and analogue. The digital supply feeds energy to the processing and RF communication sub-units and the analogue one provides energy to the sensing sub-unit. In [36], the authors presented a novel design of a WSN for mine safety, called WMSS. In WMSS, the underground node deployment satisfied the principle of minimum coverage (using minimum number of nodes while ensuring coverage). Sensor nodes consist of a gas-sensing unit, a micro-embedded processor from ATMEL and a wireless communication module operating on the 2:4 GHz-band. The operating system used in each sensor node is the TinyOS (Tiny Micro Threading Operating System). TinyOS is affair-driven with 2-level scheduling: task and hardware levels. Once a task is being scheduled, other tasks cannot interrupt it. TinyOS maintains a task-scheduling table based on a simple first-in-first-out (FIFO) algorithm. When the table has no entries, the sensor node is switched into sleep mode unless an affair awakes it. The system in [36] has implemented an energy- saving strategy that is based on a dynamic energy consumption control and management through employing efficient and dynamic/adaptive voltage monitoring mechanisms.

The authors in [37] proposed a framework for residential fire detection system based on interval-message-ration metric. They showed that their framework is applicable for any disaster recovery situations. In [37], an alarm system was proposed to assist with fire-fighting operations. In this system, temperature, humidity and light sensors are considered in an operating environment that cannot be easily accessed. The proposed design accounted for the scenario, where sensor devices can be destroyed by the fire. The authors in [38] developed a GPS-based WSN for realistic fire detection in forests, where each senor device contains a thermometer. The authors proposed a dynamic routing protocol, as sensor nodes might be destroyed by fire. They also indicated that deploying 3 sensor nodes to monitor a given location is sufficient for accurate fire detection.

The system model in [39] illustrated the design and implementation of a WSN for smoke detection. The objective of the implemented WSN is to provide timely alerts and put off all smoke sensor units in the WSN when at least one smoke sensing device goes into off mode. Each sensor node consists of a PIC micro-controller (the software elements consist of programs and codes that are uploaded into the PIC), temperature and smoke sensors, limited-power battery and RF radios. The proposed WSN was tested and showed a proper operation. The authors in [40] developed a WSN that detects temperature, humidity and smoke based on multi-parameter coincidence technique. A multi-parameter wireless node with intelligent fire-detection capability is developed. Such capability integrates signal detection and signal processing within the detector without using the RF communication circuitry. This enhances signal detection accuracy. According to the proposed WSN in [40], each sensor is equipped with three sensing devices: temperature detection device, humidity detection device and photoelectric-type smoke detection device. The operating principle of the photoelectric device is based on the fact that smoke particles of the same wavelength re-radiate energy when smoke particles interact with light. Smoke particle signal detection is accomplished using a transmitting and receiving pair of infrared light diodes. The transmitting infrared diode sends light pulses to the detection area. When no smoke particles are present in the detection area, the emitted light will not reach the receiving diode because of the light resistance shading plate. In case of fire, smoke particles will exist in the detection area and hence the infrared light will be scattered by the existing smoke particles, resulting in reaching the photodiode and exciting a current signal that increases with the intensity of smoke. However, this scheme may trigger false alarms due to some other particles other than smoke particles.

The work in [41] proposed a wireless gas sensor network (WGSN) for monitoring and identifying of hazard gases in buildings, residential and industrial facilities, with the use of solid-state types of sensors with metal oxides (e.g., tungsten-oxide). A heating element is used in each sensor node to regulate the sensor temperature, since this gas sensor exhibits different gas response characteristics at different temperature ranges. Solid-state sensors are adaptable and long lived. These sensors have the ability of detecting very low to very high LPG concentration, making them suitable for identifying/monitoring poisoning and dangerous/explosive gas levels. However, a solid-state-based WSN has poor selectivity toward a gas to be detected. Worse yet, solid-state-based sensors consume high power, which requires a bulky and continuous power supply. The paper in [42] developed a new wireless sensor device for CO_2 and LPG level monitoring using spectroscopy absorption mechanisms with C-band type laser diodes. The proposed sensor unit integrates optical fiber sensor devices with wireless networking capabilities, which can realize a distributed remote gas monitoring on real-time basis using low-cost/low-power RF radios CC2500. In [43], a WSN was used for gas and fire detection in indoor and outdoor environments. In this work, the authors proposed a monitoring system for safety and environmental monitoring scenarios. The developed system includes: sensing mechanisms, over-the-air programming (OTAP), gas detection (for oxygen (O_2), methane (CH_4), temperature level, carbon monoxide (CO), carbon dioxide (CO_2), flammable alcohols and nitrogen dioxide (NO_2)), GPRS unit (capable of providing SMS messaging or direct calls, FTP database to upload and download data, Internet connection using TCP/UDP), GPS unit (capable of providing global time-synchronization, longitude, latitude and velocity), e-mail server (capable of generating e-mail updates on regular basis associated with the operation of the network).

In [44], the authors developed a centralized monitoring and early gas-leakage detection system using WSN nodes that are based on a microcontroller and control system. The developed system uses XBee PRO S2B nirkable devices to enable wireless networking to send the sensed data to a PC and software-integrated Visual Basic. In [45], the authors have developed a WSN for detecting gas leakage by pacing sensors around the gas tube and its distribution line. They used MQ-6 as a gas sensor and wireless Bluetooth module HC-05. Upon detecting gas leaks, the sensor wirelessly sends the data to an Arduino module that activates an explosion prevention system. In [46], a WSN is designed to detect, monitor and control the hazardous gas hydrogen sulfide in an industrial transportation system. Each sensor node in this design consists of an advanced PIC 18F4550 microcontroller along with sensing and signal conditioning capabilities and an IEEE 802.15.4 slandered-based ZigBee module. The sensed data is sent to a sink node that can close the main valve of the gas source. A wireless sensor actuator network (WSAN) was developed in [47] to monitor, detect and control the leakage of hazardous gases in an industrial transportation system to prevent catastrophic accidents. Each sensor-actuator node in this design is equipped with a PIC 18F4550 microcontroller along with sensing

66

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

devices and an IEEE 802.15.4. A sink node can perform a control electromechanical action. In [48], hardware and software implementations are proposed to enhance safety and effectiveness of fire-fighting operations. The developed system contains three main subsystems: (1) a WSN referred to as SmokeNet, which is the main part of the system, (2) a display subsystem that is head mounted for each firefighter referred to as FireEye and (3) a command system referred to as eICS. The eICS is a visual display that shows information including location of fire-fighters and their biometric data. TinyOS is used to implement SmokeNet, which utilizes Crossbow wireless smoke and temperature sensors. When no alert is generated, sensor devices sense the status of the environment every 10 seconds and transmit the sensed data along with their battery level to a central command node every 5 minutes. Once a fire is detected, a sensor device generates alarm messages that switch the entire WSN to the alert state. In such state, each sensor node senses the occurrence of fire in a time scale of 5 seconds and transmits the collected data to a control unit every 2 minutes in case of no fire detection to verify that the network is still correctly operating (i.e., alive) [48]. The works in [49]-[54] also used the same design methodology of those proposed in [46]-[47] and [48].

Very few works have been proposed to design IoT-based LPG early detection systems (e.g., [55]-[57]). In [55] and [56], the authors modified an existing safety system that is already implemented in industries, homes and offices by designing microcontroller-based LPG and propane gas detecting and alerting system. The LPG concentration is continuously monitored and reported to an LCD display. If the LPG level exceeds the accepted level, the system immediately generates an alarm message (e-mail) and sends it through the Internet using ARM development board to the person in charge. In [57], the authors developed an LPG leakage detector that is connected to the Internet (IoT-capable) using ESP module and Arduino controller. This system is used to detect LPG leakage from cylinders and alert the users through IoT software. However, the aforementioned proposed IoT-based systems were designed as stand-alone sensing systems that do not fully utilize the WSN technology. Table 2 summarizes the various surveyed WSN/IoT approaches in terms of IoT support, wireless transmission strategy, deployment strategy, networking support and the type of used sensing device.

Table 2. Comparison between the various surveyed WSN/IoT approaches.

System Design	IoT Support	Transmission Module	Deployment Strategy	Networking Support	Sensing Device
Multi-level cluster-based WSN [23]-[24]	Partially (Integration with Internet services)	Cognitive Radio Module	Pre-deployed cluster network	Yes	Gas sensor MQ6
WGSN in [25],[27]	Partially (Integration with Internet services)	ZigBee module (IEEE 802.15.4 protocol) and GSM/GPRS connection for sink	Pre-deployed flat network	Yes	2D semiconductor sensor
Visual WSN in [26]	No	Long Range-Low Power Data Radio Modem LR96	Pre-deployed flat network	Yes	Visual and gas leakage sensors
Stand-alone gas-detection wireless systems [28]-[30]	No	A detector with wireless capability module and a wireless hand-held remote module	Pre-deployed flat network	No (Stand-alone warning systems)	2D semiconductor sensor
WSN prototype for hydrogen gas-leak detection [31]	No	Generic 433 MHz transmitter/reciever wireless module	Pre-deployed flat network	Yes (10 wireless nodes)	Field-effect-transistor (FET) sensors with a diode thermometer
Gas leak detection-location system based on ZigBee [32]	No	2.4 GHz IEEE 802.15.4/ZigBee Module (CC2430)	Hybrid (Pre-deployed and on-the-spot dynamic-mobile nodes)	Yes	Gas sensors MQ4 and MQ6
Siren system: a context-aware WSN [34]	No	2.4-WiFi capable PDA	Hybrid pre-deployed and on-the-spot (tacit networking between fire-fighters and pre-existing nodes inside building)	Yes	Generic temperature sensor, gas sensor, and smart dust sensor

"A Survey on Wireless Sensor Network-Based IoT Designs For Gas Leakage Detection and Fire-Fighting Applications", H. Bany Salameh, M. Dhainat and E. Benkhelifa.

Energy-efficient LPG WSN system [35]	No	ETRX357 ZigBee transceiver (IEEE 802.15.4 protocol)	Pre-deployed flat topology	Yes	Catalytic gas sensor (e.g., DTK-2, NAP-66A)
A mine safety system based on WSN (WMSS) [36]	No	2.4 GHz ZigBee module	Pre-deployed nodes and movable nodes	Yes	Generic gas sensors
Residential fire detection system based on interval-message-ration [37]	No	A Zigbee-module based on IEEE802.15.4	Pre-deployed (ten sensor TIP710CM motes)	Yes	Generic temperature, humidity and light sensors
Firementor: A GPS-based WSN for realistic fire detection in forests [38]	No	RF transceiver CC1000Chipcon (433MHz)	Pre-deployed (deploying 3 sensors in a given location is sufficient for fire detection)	Yes	Electronic thermometer sensor
A wireless smoke /fire detection system [39]	No	FSIOOOA 315 MHz Wireless Radio Transmitting Module	Pre-deployed	Yes	Photoelectric smoke and thermometer sensors
a WSN based on multi-parameter coincidence technique [40]	No	CC2420 Series 2.4 GHz RF Transceiver (IEEE 802.15.4 protocol)	Pre-deployed	Yes	The intelligent temperature and humidity sensor chip SHT11
A WGSN for monitoring and identifying of hazard gases [41]	No	2.4 GHz ZigBee module (IEEE 802.15.4 protocol)	Pre-deployed flat topology	Yes	Solid-state types of sensors with metal oxides (e.g., tungsten-oxide)
Wireless sensor node gas level monitoring [42]	No	The CC2500 module:2.4 GHz transceiver designed for very low-power wireless applications	Pre-deployed	No (Single-node design)	The sensing unit includes laser current and temperature controller, laser diode, thermoelectric cooler, gas cell, photo diode and analogue-digital converter
WSN for fire emergency and gas detection [43]	Partially (Integration with Internet services)	2.4-GHz ZigBee module (IEEE 802.15.4 protocol) and GPRS connection for sink	Pre-deployed	Yes	O_2 sensor (SK-25), methane sensor, temperature sensor, CO/CO_2 sensor, NO_2 sensor (MiCS-2710)
Early detection of LPG gas leakage system [44]	No	2.4 GHz ZigBee Module (XBee PRO S2B) GPRS connection for monitoring node (sink)	Pre-deployed	Yes	Gas sensor MQ-4
LPG gas leak detection and automatic gas regulator system using Arduino [45]	No	Wireless Bluetooth module HC-05	Pre-deployed	No (Single-node design)	Gas sensor MQ-6
Wireless Sensor Actuator Network (WSAN) [46]-[54]	No	IEEE 802.15.4 slandered-based ZigBee module	Pre-deployed	Yes	Gas sensor [49]-[51] Temperature sensor Smoke sensor environmental

68

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

					temperature, air pressure around the firefighter [52]-[56]
IoT-based gas leakage monitoring system [55]	Yes	Raspberry pi : a single-board computer with wireless LAN and Bluetooth	Pre-deployed of single-device that sends an alert message (e-mail) to the authorized person *via* the Internet	No (Standalone device with IoT capability)	Gas sensor MQ-6
IoT-based gas leakage monitoring system [56]	Yes	Raspberry pi : a single-board computer with wireless LAN and Bluetooth	Pre-deployed of single-device that sends an alert message (Email) to the authorized person via the Internet	No (Standalone device with IoT capability)	Gas sensor MQ-2
IoT-based LPG leakage detector using ESP module and Arduino controller [57]	Yes	ESP module	Pre-deployed of single devices, where the ESP module is used as Wi-Fi module to connect the with IoT	No (Standalone device with IoT capability)	Gas sensor MQ-6

5. DISCUSSION AND OPEN RESEARCH PROBLEMS

As demonstrated in the previous section, large number of WSN systems have been designed for fire-fighting, gas monitoring and detection applications based on different wireless communication technologies (e.g., WIFI, Cognitive radio, IEEE 802.15.4). However, most of these designs did not fully exploit the capabilities of IoT systems and WSN technology and did not consider their unique functionalities and requirements. On the other hand, existing IoT-based LPG-detection systems were designed as stand-alone systems that did not integrate the WSN technology. Specifically, when IoT is considered, the sensor nodes can either directly transmit their sensed data immediately or periodically to the Internet (where a server can analyze the sensed data and take action) or operate as a regular sensor by forwarding their sensed data to the sink node. The sink node can be utilized by the IoT system through communicating the gathered data to the Internet. The developed communication protocols, hardware designs and processing mechanisms for WSNs cannot be directly implemented in IoT-based WSNs. Therefore, new designs and communication protocols are needed to fully utilize the IoT functionalities and capabilities in WSNs such that network performance is improved (i.e., delay, throughput, energy consumption, connectivity …etc.).

Specifically, many interesting open design issues are still to be addressed. Considering the IoT capability of allowing sensor nodes to directly communicate their sensed information to the Internet along with their communication with the sink node is quite promising, but their design assumptions and feasibility should be carefully studied. Research should emphasize also on how to communicate the sensed data such that performance guarantees are provided. Moreover, utilizing cloud-computing and storage is very promising in improving IoT-based WSN performance in terms of energy, delay and processing overhead. In conclusion, huge research efforts has been carried out on providing WSN system architectures, but none of them is totally satisfactory in terms of fully utilizing the IoT technology. Therefore, more research efforts need to be carried out to provide comprehensive IoT-based WSN solutions and designs that are well-suited to gas-leakage detection, monitoring and fire-fighting applications while exploiting the additional features and functionality of the IoT architecture and WSN technology.

6. CONCLUSIONS

A comprehensive overview of existing and emerging system designs for gas-leakage monitoring and detection using IoT-based WSNs is provided in this survey. We highlighted the main features of such networks and showed the main differences between event-driven and data-collection networks. With the huge advancement of sensing technologies and the emerging wave of large-scale IoT deployment, gas-leakage detection and monitoring techniques are analyzed from the point-of-view of precision, system architecture, simplicity, robustness and energy consumption issues. Furthermore, we summarized the state-of-the-art designs of WSNs for gas-leakage and fire-fighting applications.

"A Survey on Wireless Sensor Network-Based IoT Designs For Gas Leakage Detection and Fire-Fighting Applications", H. Bany Salameh, M. Dhainat and E. Benkhelifa.

We showed that these WSN systems differ in their performance objectives and most of them did not exploit the potential benefits of IoT systems such as allowing sensor nodes to communicate their sensed data directly to the Internet and utilizing cloud processing and storage capabilities. We also showed that the existing IoT-based systems for LPG leakage detection were designed to operate as stand-alone sensing systems that did not exploit the WSN technology.

ACKNOWLEDGEMENTS

REFERENCES

[1]   Y. Kuo, C. Li, J. Jhang and S. Lin, "Design of a Wireless Sensor Network-based IoT Platform for Wide Area and Heterogeneous Applications," IEEE Sensors Journal, vol. 18, no. 12, pp. 5187-5197, 2018.

[2]   M. Kocakulak and I. Butun, "An Overview of Wireless Sensor Networks Towards Internet of Things," Proc. of the IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, pp. 1-6, 2017.

[3]   H. Bany Salameh, S. Almajali, M. Ayyash and H. Elgala, "Spectrum Assignment in Cognitive Radio Networks for Internet-of-Things Delay-Sensitive Applications Under Jamming Attacks," IEEE Internet of Things Journal, vol. 5, no. 3, pp. 1904-1913, June 2018.

[4]   A. Kumar, K. Ovsthus and L. Kristensen, "An Industrial Perspective on Wireless Sensor Networks- A Survey of Requirements, Protocols and Challenges," IEEE Communications Surveys & Tutorials, vol. 16, no. 3, pp. 1391-1412, 2014.

[5]   M. Ayaz, M. Ammad-uddin, I. Baig and E. M. Aggoune, "Wireless Sensors Civil Applications, Prototypes and Future Integration Possibilities: A Review," IEEE Sensors Journal, vol. 18, no. 1, pp. 4-30, Jan., 2018.

[6]   H. Bany Salameh, T. Shu and M. Krunz, "Adaptive Cross-layer MAC Design for Improved Energy-Efficiency in Multi-Channel Wireless Sensor Networks," Ad Hoc Networks Journal, vol. 5, no. 6, pp. 844-854, Aug. 2007.

[7]   S. M. Kamruzzaman, M. Jaseemuddin, X. Fernando and P. Moeini, "Wireless Positioning Sensor Network Integrated with Cloud for Industrial Automation," Proc. of the 42nd IEEE Conference on Local Computer Networks (LCN), Singapore, pp. 543-546, 2017.

[8]   S. A. I. Quadri and P. Sathish, "IoT Based Home Automation and Surveillance System," International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, pp. 861-866, 2017.

[9]   F. Wang and J. Liu, "Networked Wireless Sensor Data Collection: Issues, Challenges and Approaches," IEEE Communications Surveys & Tutorials, vol. 13, no. 4, pp. 673-687, 2011.

[10]   F. Lin, C. Chen, N. Zhang, X. Guan and X. Shen, "Autonomous Channel Switching: Towards Efficient Spectrum Sharing for Industrial Wireless Sensor Networks," IEEE Internet of Things Journal, vol. 3, no. 2, pp. 231-243, April 2016.

[11]   J. Ma, J. Wang and T. Zhang, "A Survey of Recent Achievements for Wireless Sensor Networks Testbeds," International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Nanjing, pp. 378-381, 2017.

[12]   M. Hani, H. Bany Salameh, Y. Jararweh and A. Bousselham, "Traffic-Aware Self-Coexistence Management in IEEE 802.22 WRAN Systems," Proc. of the 7th IEEE GCC Conference and Exhibition (GCC), Doha, pp. 507-510, 2013.

[13]   A. Boulogeorgos, H. Bany Salameh and G. K. Karagiannidis, "Spectrum Sensing in Full-Duplex Cognitive Radio Networks under Hardware Imperfections," IEEE Transactions on Vehicular Technology, vol. 66, no. 3, pp. 2072-2084, March 2017.

[14]   T. Shu, H. Bany Salameh and M. Krunz, "Cross-layer Optimization of a CSMA Protocol with Adaptive Modulation for Improved Energy Efficiency in Wireless Sensor Networks," Proc. of the IEEE GLOBECOM Conference, San Francisco, Dec. 2006.

[15]   S. E Jero and A. B. Ganesh, "PIC18LF4620 Based Customizable Wireless Sensor Node to Detect Hazardous Gas Pipeline Leakage," Proc. of the International Conference on Emerging Trends in Electrical and Computer Technology, pp.563-566, 2011.

[16]   A. Somov, D. Spirjakin, A. Spirjakin, A. Baranov, V. Sleptsov and R. Passerone, "Safe Delivery of Sensed Data in Wireless Sensor Networks for Gas Leak Detection: A Boiler Facility Scenario," Proc. of the 26th European Conference on Solid-State Transducers (Eurosensors'12), 2012.

[17]   M. Akhondi, A. Talevski, S. Carlsen and S. Petersen, "Applications of Wireless Sensor Networks in the Oil, Gas and Resources Industries," Proc. of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), pp. 941-948, April 2010.

[18]   M. Rakesh and S. Dagadi, "Implementation of Wireless Gas Leakage Detection System," Proc. of the 6th International Conference on Sensing Technology (ICST), 2012.

[19]   M. Rahmati, H. Yazdizadeh and A. Yazdizadeh, "Leakage Detection in a Gas Pipeline Using Artificial Neural Networks Based on Wireless Sensor Network and Internet of Things," Proc. of the IEEE 15th International Conference on Industrial Informatics (INDIN), Emden, pp. 659-664, 2017.

[20]   H. Li, Y. Yang, X. Qiu, Z. Gao and G. Ma, "Gravitation-Based 3-D Redeployment Schemes for the Mobile Sensors and Sink in Gas Leakage Monitoring," IEEE Access, vol. 5, pp. 8545-8558, 2017.

[21]   D. Spirjakin and A. M. Baranov, "Internet Connected Wireless Combustible Gas Monitoring System for Apartment Buildings," Federated Conference on Computer Science and Information Systems (FedCSIS), Prague, pp. 873-876, 2017.

[22]   S. Ali et al., "SimpliMote: A Wireless Sensor Network Monitoring Platform for Oil and Gas Pipelines," IEEE Systems Journal, vol. 12, no. 1, pp. 778-789, March 2018.

[23]   H. Bany Salameh, M. F. Dhainat, A. Al-Hajji, R. Aqeli and M. Fathi, "A Two-Level Cluster-Based Cognitive Radio Sensor Network: System Architecture, Hardware Design, and Distributed Protocols," IEEE International Conference on Cloud Engineering, Tempe, AZ, pp. 287-292, 2015.

[24]   H. Bany Salameh and M. Dhainat, "Event-driven Hybrid MAC Protocol for a Two-tier Cognitive Wireless Sensor Network: Design and Implementation," International Journal of High Performance Computing and Networking (IJHPCN), vol. 9, no. 4, 2016.

[25]   A. Somov, A. Baranov, A. Savkin, D. Spirjakin, A. Spirjakin and R. Passerone, "Development of Wireless Sensor Network for Combustible Gas Monitoring, Sensors and Actuators," Physical, vol. 171, no. 2, pp. 398-405, 2011.

[26]   M. S. Tadpatrikar and J. G. Rana, "Automated Gas Leakage Detection System with Visual Data Monitoring Using Wireless Visual Sensor," Proc. of the SPIT-IEEE Colloquium and International Conference, Mumbai, India, 2011.

[27]   N. Kurata, B. Spencer and M. Ruiz-Sandoval, "Application of Wireless Sensor Mote for Building Risk Monitoring," [Online], Available: http://www.unl.im.dendai.ac.jp, 2003.

[28]   Patent US20100141276- Methods for Detecting Unauthorized Access to Oil and Gas- 2010.

[29]   Wireless Gas Leak Detector (Model EL-2762).

[30]   Patent US20070000310 - Leak Detection System with Wireless Remote Unit-2007.

[31]   S. Nakano, Y. Goto, K. Yokosawa and K. Tsukada, "Hydrogen Gas Detection System Prototype with Wireless Sensor Networks," IEEE Sensors Conference, Irvine, CA, 2005.

[32]   D. Chengjun, L. Ximao and D. Ping, "Development on Gas Leak Detection and Location System Based on Wireless Sensor Networks," Proc. of the 3rd International Conference on Measuring Technology and Mechatronics Automation, 2011.

[33]   R. Suganya and P. Suseendhar, "Online Monitoring of Green House Gas Leakage in Industries," IJREAT International Journal of Research in Engineering and Advanced Technology, vol. 1, no. 6, Jan. 2014.

[34]   X. Jiang, N. Y. Chen, J. I. Hong, K. Wang, L. Takayama and J. A. Landay, "Siren: Context-aware Computing for Firefighting," Proceedings of the 2nd International Conference on Pervasive Computing (Pervasive 2004), Vienna, Austria, April 2004.

[35]   A. Somov, A. Baranov, A. Savkin, M. Ivanov, L. Calliari, R. Passerone, E. Karpov and A. Suchkov, "Energy-Aware Gas Sensing Using Wireless Sensor Networks," Wireless Sensor Networks, Springer, Berlin, Heidelberg, pp. 245-260, 2012.

"A Survey on Wireless Sensor Network-Based IoT Designs For Gas Leakage Detection and Fire-Fighting Applications", H. Bany Salameh, M. Dhainat and E. Benkhelifa.

[36]   W. Tan, Q. Wang et al., "Mine Fire Detection System Based on Wireless Sensor Network," Proceedings of the International Conference on Information Acquisition (ICIA '07), Seogwipo-si, South Korea, 2007.

[37]   Y. Lim, S. Lim et al., "A Fire Detection and Rescue Support Framework with Wireless Sensor Networks," Convergence Information Technology, Gyeongju, South Korea, 2007.

[38]   V. Vescoukis, T. Olma et al., "Experience from a Pilot Implementation of an "In- Situ" Forest Temperature Measurement Network," Proceedings of Personal, Indoor and Mobile Radio Communications (PIMRC2007), Athens, Greece ,2007.

[39]   M. Lipu, T. Karim, M. Rahman and F. Sultana, "Wireless Security Control System and Sensor Network for Smoke and Fire Detection," Proc. of the IEEE International Conference on Advanced Management Science (ICAMS), pp.153-157, Chengdu, China, July 2010.

[40]   L. Shixing, T. Defeng and Z. Yongming, "Multi-parameter Fire Detection Based on Wireless Sensor Network," Proc. of the IEEE International Conference on Intelligent Computing and Intelligent Systems, pp.203,206, Nov. 2009.

[41]   P. Jain and R. Kushwaha, "Wireless Gas Sensor Network for Detection and Monitoring of Harmful Gases in Utility Areas and Industries," Proc. of the 6th International Conference on Sensing Technology (ICST), pp. 642-646, Dec. 2012.

[42]   F. Dian, "Development of Novel Gas Detection Wireless Sensor Node," Proc. of Spring Congress on Engineering and Technology (S-CET), pp.1-3, May 2012.

[43]   A. Qandour, D. Habibi and I. Ahmad, "Wireless Sensor Networks for Fire Emergency and Gas Detection," Proc. of the 9th IEEE International Conference on Networking, Sensing and Control (ICNSC), pp.250-255, April 2012.

[44]   M. Kusriyanto, M. Firdaus, A. Yulianto and S. Kurniawan, "Early Detection of LPG Gas Leakage Based on Wireless Sensor Networking," MATEC Web of Conferences, 2018.

[45]   L. Dewi and Y. Somantri, "Wireless Sensor Network on LPG Gas Leak Detection and Automatic Gas Regulator System Using Arduino," International Symposium on Materials and Electrical Engineering (ISMEE), 2017.

[46]   P. V. Mane-Deshmukh, D. M. Adat, B. P. Ladgaonkar and S. K. Tilekar, "Designing of an Embedded System for Wireless Sensor Network for Hazardous Gas Leakage Control for Industrial Application," Journal on Embedded Systems, vol. 6, no. 2, pp. 1-9, 2018.

[47]   P. V. Mane-Deshmukh, D. M. Adat, B. P. Ladgaonakar and S. K. Tilekar, "Monitoring and Control of Gas Leakages of Industrial Sector Using PIC 18F4550, ZigBee and Wireless Sensor Actuator Network," Journal on Electronics Engineering, vol. 8, no. 3, pp. 5-13, 2018.

[48]   D. Steingart, J. Wilson, A. Redfern, P. Wright, R. Romero and L. Lim, "Augmented Cognition for Fire Emergency Response: An Iterative User Study," Proceedings of the 1st International Conference on Augmented Cognition, Las Vegas, NV, July 2005.

[49]   S. Unnikrishnan, M. Razil, J. Benny, S. Varghese and C. V. Hari, "LPG Monitoring and Leakage Detection System," International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, pp. 1990-1993, 2017.

[50]   Z. A. Khan, E. Pignaton de Freitas, T. Larsson and H. Abbas, "A Multi-Agent Model for Fire Detection in Coal Mines Using Wireless Sensor Networks," Proc. of the 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp.1754-1761, July 2013.

[51]   P. Kulakowski, E. Calle and J.-L. Marzo, "Sensors-Actuators Cooperation in WSANs for Fire-Fighting Applications," Proc. of the 6th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp.726-732, 2010.

[52]   H. Liu, J. Li, Z. Xie, S. Lin, K. Whitehouse, J. Stankovic and D. Siu, "Automatic and Robust

         readcrumb System Deployment for Indoor Firefighter Applications," Proc. of the 8th International Conference on Mobile Systems, Applications and Services (MobiSys'10), New York, NY, USA, pp. 21-34, 2010.

[53]   A. May, V. Mitchell and J. Piper, "A User Centred Design Evaluation of the Potential Benefits of Advanced Wireless Sensor Networks for Fire-in-Tunnel Emergency Response," Fire Safety Journal, vol. 63, pp. 79-88, 2014.

[54]   A. Chunlei and T. -G. Andreas, "Applying Wireless Sensor Networks in Fire Fighting," Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 97, pp. 386-395, 2012.

[55]   R. Pandey, M. Verma, L. Sahu and S. Deshmukh, "Internet of Things (IoT) Based Gas Leakage Monitoring and Alerting System with MQ-6 Sensor," International Journal of Creative Research Thoughts (IJCRT), vol. 6, no. 1, 2018.

[56]   R. Pandey, M. Verma and L. Sahu, "Internet of Things (IoT) Based Gas Leakage Monitoring and Alerting System with MQ-2 Sensor," International Journal of Engineering Development and Research, vol. 5, 2017.

[57]   V. Naren, P. Indrajith, R. Prabhu and C. Ganesh, "Intelligent Gas Leakage Detection System with IoT Using ESP 8266 Module," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 7, no. 12, December 2018.

ملخص البحث:

لقــد أعطــت التطــورات المتســارعة فــي مجــال شــبكات المجســات اللاســلكية دافعــاً قويـاً لتوظيــف إنترنــت الأشــياء علــى نطــاق واســع فــي العديــد مــن التطبيقــات والخــدمات. أحــد هــذه التطبيقــات يتمثـل فــي اســتخدام عُقَــد المجســات اللاســلكية فــي رصــد تسـرّب الغــاز والكشــف عنــه. والجــدير بالــذكر أنّ مــن شــأن مثــل هــذه التطبيقــات لإنترنــت الأشــياء أن تــوفر حمايــة أفضــل لرجــال مكافحــة الحرائــق وتــؤمّن شــيئأ مــن الســلامة والإنــذار المبكـر في الوقت المناسب للأفراد والشركات والمؤسسات على حدّ سواء.

فــي هــذه الورقــة البحثيــة، نســلّط الضــوء علــى الخصــائص التــي تتفــرّد بهــا شــبكات المجســات اللاســلكية، وننــاقش المتطلبــات التصــميمية لتلــك الشــبكات فيمــا يتعلــق بتطبيقــات رصــد تسـرّب الغــاز والكشــف عنــه. كمــا ننــاقش الاختلافــات الرئيســية بــين حلــول شــبكات المجسّــات اللاســلكية القائمــة علــى جمــع البيانــات وتلــك القائمــة علــى الكشــف عن الاحداث.

مــن جهــة أخــرى، تتضــمن هــذه الورقــة مراجعــة تفصــيلية للدراســات والبحــوث الســابقة التــي تــم إنجازهــا فــي مجــال تــوفير حلــول عبــر شــبكات المجســات اللاســلكية للتطبيقــات المتعلقة برصد تسرّب الغاز والكشف عنه.

73

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

A COMPARATIVE STUDY OF DCT AND DWT IMAGE COMPRESSION TECHNIQUES COMBINED WITH HUFFMAN CODING

Ashraf Maghari

ABSTRACT

Image compression techniques have been widely used to store and transmit data which requires storage space and high transfer speed. The explosive growth of high-quality photos leads to the requirement of efficient technique to store and exchange data over the internet. In this paper, we present a comparative study to compare between the Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) algorithms in combination with Huffman algorithm; DCT-H and DWT-H. The comparison is based on five factors: Compression Ratio (CR), Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) and compression/decompression time. The experiments are conducted on five BMP gray-scale file images. We found out that DWT-H coding is comparable to DCT-H coding in term of CR and outperforms DCT-H in terms of MSE, PSNR and SSIM. The CR average results of the five test images for DCT-H and DWT-H are 2.36 and 3.17, respectively. Moreover, DCT-H has the average results of MSE = 13.19, PSNR = 37.15 and SSIM = 0.76, while WDT-H has the average results of MSE = 4.54, PSNR = 42.5 and SSIM = 0.85. On the other hand, DCT-H outperforms DWT-H in term of execution time for compression and decompression. DCT-H has an average compression time of 0.358s and an average decompression time of 0.122s, while WDT-H has 2.38s compression time and 2.13s decompression time.

1. INTRODUCTION

In the last decades, the demand for image compression has increased, particularly after the great development of camera devices and the proliferation of high-quality image and video exchange over the internet [1]. The applications based on images, such as medical imaging, cameras and video-on-demand systems contain large amounts of data to transmit. The main idea behind image compression is to reduce the size of the image in order to minimize the storage space and increase the transmission speed [2]. Data Compression (DC) is a technique that transforms the original data to its compact form by the recognition and utilization of patterns existing in the data. It should be able to inverse the data very approximately to original data [3]. DC techniques are crucially used in many real time applications, like satellite imagery, Geographical Information Systems (GISs), graphics, Wireless Sensor Networks (WSNs), …etc. For example, an image without compression contains 1024 pixels of 24 bit with a size of 3MB. The image needs a transmission time of 7 minutes with an ISDN line and 64 Kbit/s. If the image is compressed at 10:1 compression ratio, its size drops to 300 KB and needs below 6 seconds transmission time [4]. The purpose of compression is to eliminate data redundancy and irrelevancy in order to decrease the storage and transmission costs while maintaining good quality.

There are two different categories of image compression techniques; lossy and lossless techniques. Lossy compression is a technique that transforms the original data into more efficient data and cannot reconstruct the original data without errors. It is also called transform coding [3]. Lossless compression is a technique that processes the original data without losing any information [5]. Examples of lossy techniques include Discrete Hartley Transform (DHT), Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), …etc. [4]. Lossless compression techniques include Huffman coding, LZW, arithmetic coding, …etc.

Transform coding converts the input data into another kind of representation in which the transformed values (coefficients) are encoded by compression techniques. DCT and DWT are the most widely used

---

A. Maghari is with Faculty of Information Technology, Islamic University of Gaza, Palestine. Email: amaghari@iugaza.edu.ps

transform coding techniques which have the ability to compress data using a smaller number of coefficients [3]. The major drawbacks, however, of DCT are preface of fake contouring effects and blocking artifacts at higher compression. Similarly, DWT requires huge computational resources.

Mostly, transform-based image compression algorithms follow three step processes: transformation, quantization and entropy encoding. The quantization may be scalar or vector and the quantized transformed coefficients are entropy-coded [6]. On the other hand, Huffman coding as a lossless algorithm has a good compression ratio and a fast compression time. A previous study showed that Huffman coding is better than RLE and Delta encoding techniques in term of compression time [7]. Huffman replaces fixed-length code words with variable-length code words, where low-frequency symbols are expressed with longer encodings and high-frequency symbols are expressed with shorter encodings. In this paper, we have analyzed and implemented DCT image compression technique combined with Huffman coding (DCT-H) and DWT image compression technique combined with Huffman coding (DWT-H). The two combined techniques have been evaluated using different performance metrics.

There are many comparative studies which have been conducted to compare between compression techniques using DCT and DWT [8], [9]. However, their experiments were conducted on only one or two images with different sizes and did not take into account the compression and decompression time. The main objective of this work is to empirically compare between DWT and DCT in combination with Huffman coding in terms of five factors: Compression Ratio (CR), Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) as well as compression and decompression time. This study will be helpful for upcoming researchers to approximately select and develop the required algorithms to be used in a particular situation. Our experiments are conducted using five different images with the same size (256×256). The size of (256×256) pixels is chosen to ease the implementation of DCT and WDT, where the image components are divided into (8×8) blocks for DCT implementation.

The remainder of the paper is organized as follows. We present a background in Section 2. The related work is described in Section 3. Section 4 presents the methodology used and the experimental setup. Results and discussion are explained in Section 5. Section 6 concludes our findings of this research.

2. BACKGROUND

2.1 Lossy and Lossless Image Compression

Lossless compression is commonly used with text files, where the input and output data are the same, before and after the compression process, while lossy compression is commonly used with different types of data, such as image, video and audio data. At lossy compression the input data and output data are not the same, which means that there is a loss of some data at the compression process so that when we perform the decompression process, we obtain a closer approximation of the input data [7].

2.2 Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) is one of the lossy compression approaches which are commonly applied in photo compression as jpeg compression. DCT is very approximate to Discrete Fourier Transform (DFT), but DCT includes a basis of cosine functions and real number co-efficients. Both DFT and DCT transform data from a spatial domain into a frequency domain. An inverse function is used to reconstruct the image back [10]. The basic idea of DCT is to convert a signal into basic frequency components. The image is divided into several blocks. Then, the sum of cosine functions on different frequencies can be mathematically used to express each block of an image. For example, Joint Photographic Experts Group (JPEG) [11] is a well-known compression scheme based on DCT. For DCT, the image is first transformed into an appropriate format for image compression. All the image components are divided into (8×8) blocks. Every block is encoded using discrete cosine

75

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

transformation, which is used to exploit the spatial correlation between the pixels [2]. The following formulae in Equation (1) and Equation (3) describe the transformation function and its inverse.

2.2.1 Definition of DCT

Given a function *f(i, j)* (an M*N block of an image), the 2D DCT transforms it into a new function *F(u, v)*. The general definition of the transform is:

$$F(u.v) = \frac{2C(u)C(u)}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} cos\frac{(2i+1)u\pi}{2M} * cos\frac{(2j+1)v\pi}{2N} f(i.j) \tag{1}$$

where i, u = 0, 1,..., M-1, j, v = 0, 1,..., N-1 and the constants C(u) and C(v) are defined as:

$$c(k) = \begin{cases} \frac{\sqrt{2}}{2} \; if \; k = 0 \\ 1 \; otherwise \end{cases} \tag{2}$$

The image block dimension is defined as M=N=8.

2.2.2 2D Discrete Cosine Transform (2D DCT)

The 2D transform function is defined as follows:

$$F(u.v) = \frac{C(u)C(u)}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} cos\frac{(2i+1)u\pi}{16} * cos\frac{(2j+1)v\pi}{16} f(i.j) \tag{3}$$

where i, u, j, v = 0, 1, ..., 7 and the constants C(u) and C(v) are determined by Equation (2).

2D Inverse DCT (2D IDCT):

The 2D inverse function is defined as follows:

$$f(i.j) = \sum_{i=0}^{7} \sum_{j=0}^{7} \frac{C(u)C(u)}{4} cos\frac{(2i+1)u\pi}{16} * cos\frac{(2j+1)v\pi}{16} F(u.v) \tag{4}$$

where i, u, j, v = 0, 1, ..., 7 and the constants C(u) and C(v) are determined by Equation (2).

After the transformation, most of the information is intense to a few low-frequency components. These components are then quantized in order to decrease the number of bits required to represent the image. The quantization step produces many zero components in the bit stream. Therefore, entropy encoding, such as Huffman encoding, can achieve better compression.

2.3 Image Compression by Wavelet Transform

Discrete Wavelet Transform (DWT) is another popular technique for lossy compression, which is commonly applied in photo-video compression area. The main idea of DWT is to decompose a signal into a set of basic functions which transforms a discrete time signal into a discrete wavelet representation. DWT is mostly based on two-dimensional discrete wavelet transform (2DDWT). It uses one-dimensional discrete wavelet transform (1D-DWT) row-wise to get low (L) and high (H) bands. Then, 1D-DWT can be applied column-wise to get four sub-bands, such as LL, LH, HL and HH. Furthermore, each of these four bands can then be divided into four sub-bands. Some schemes based on wavelet transform are discussed in [2].

On the other hand, the inverse of DWT (IDWT) should be capable of inversion at minimum approximately to the original signal. Equation ( ) shows the expanded and translated family of wavelets ψ [12] which is an orthonormal basis of $L^2(R)$:

$$\left\{ \psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \; \psi\left(\frac{t - 2^j n}{2^j}\right) \right\}_{(j,n)\in z^2} \tag{5}$$

where Z represents the set of integers. The scale of translations is changed along with the overall scale 2^j, so as to keep movement in the lower resolution image in proportion [12]. The simplest wavelet transform is called Haar Wavelet Transform, which forms averages and differences of a given sequence of numbers. Theory of Haar WT algorithm can be found in [13].

2.4 Lossless Entropy Coding Techniques

Coding is a technique that assigns binary digits to the transformed and quantized output. Variable-

length coding, also known as source coding, is used to have a less average length of bits per pixel for the image, in which it replaces each input symbol with a specific code word. The entropy represents the average amount of information contained per symbol in the source S. It is the average number of bits needed to represent the symbols in the source S. The entropy H(S) of an information source with alphabets $S = \{s_1, s_2, ..., s_n\}$ is:

$$H(S) = \sum_{i=1}^{n} p_i \, log_2 \frac{1}{p_i} \qquad (6)$$

where p_i is the probability that a symbol s_i will occur in S, $log_2 \frac{1}{p_i}$ indicates the amount of information contained in s_i, which corresponds to the number of bits needed to encode s_i. The most widely used entropy coding methods in data compression literature are Huffman coding and Shannon–Fano coding.

2.5 Huffman Coding

Huffman coding is a famous compression coding technique, which replaces fixed-length code words with variable-length-code words, where low-frequency symbols are expressed with longer encodings and high-frequency symbols are expressed with shorter encodings [14]. It is a type of optimal prefix code, which is widely employed in lossless data compression. Huffman coding is uniquely decodable and consists of two components such as constructing Huffman tree from input sequence and traversing the tree to assign codes to characters. It is noticed that the two nodes at the same level of the tree will have the same code lengths. Huffman coding is still popular because of its simpler implementation, faster compression and lack of patent coverage. Several compression methods, like Deflate, JPEG, MP3, …etc. use Huffman code as the back-end technique [3].

3. RELATED WORK

There are many studies which have been conducted on image compression techniques. Hnesh and Demirel [15] proposed a new hybrid model that integrated DWT and DCT-SVD techniques. They applied their model on jpeg images and compared their results to other experiments that they performed on original DCT and DCT-SVD separately. They reported that their model results were better than DCT and DCT-SVD in term of compression ratio.

Other efforts [16] present a new framework for compression of medical images based on compressive sensing (CS). The researchers proposed a framework that combined the discrete cosine transform (DCT) as well as the discrete wavelet transform (DWT) with CS. Their framework was applied on CT and MRI medical images. Their results showed that the CS-and DWT-based compression technique performs better than other compression techniques in terms of compression ratio and PSNR for MRI and CT images.

Sharma and Kaur [17] proposed a hybrid model, using DWT, DCT and Huffman coding, for the purpose of medical image compression. Their hybrid model aims to achieve higher compression rates by first applying DWT and DCT on RGB components. Then, quantization was applied to calculate probability index for each unique quantity in order to find out the unique binary code for each unique symbol to encoding. Huffman coding was then applied on the quantized components. Their results showed that their hybrid model can effectively improve the compression ratio and PSNR.

All of the above studies applied different algorithms in image compression based on several factors to achieve a higher compression ratio. There are also some comparative studies that have been conducted to compare compression techniques. Barbhuiya et al. [8] conducted a comparative study on image compression using DCT (Discrete Cosine Transform) and DWT (Discrete Wavelet Transform). They also presented a compression algorithm using DWT and Inverse DWT and applied it on two image formats; JPEG (Joint Photographic Experts Group) and PNG (Portable Network Graphics) color images. Their results showed that the DWT technique outperforms the DCT technique in terms of Compression Ratio, Mean Square Error (MSE) and Peak Signal to Noise Ratio (PNSR).

Saroya and Kaur [9] proposed a comparative study between DCT and DWT algorithms based on the parameters mean square error (MSE) and peak signal to noise ratio (PSNR). Their experiments were conducted on one image (Lena) without showing any detailed results related to PSNR or MSE. Most of reviewed comparative studies did not report detailed results and did not take into account the compression and decompression time. This work, however, compares DWT and DCT in combination with Huffman coding in terms of four factors; Compression Ratio (CR), Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) as well as compression and decompression time.

4. METHODOLOGY AND EXPERIMENTAL SETUP

In this section, we present the idea of image reconstruction, the compression and decompression models, discuss the performance metrics and then explain the experimental setup.

4.1 Image Compression

The principle of image compression algorithms is reducing the redundancy in the image data and (or) producing a reconstructed image from the original image with the introduction of an error that is insignificant to the intended applications. The aim here is to obtain an acceptable digital image representation, while preserving the essential information contained in that particular dataset [18].

4.2 Compression/Decompression Model (DCT-H)

Due to our methodology, we applied lossy and lossless algorithms on some test images. Figure 1 shows the framework followed to implement our compression/decompression model for DCT-H.
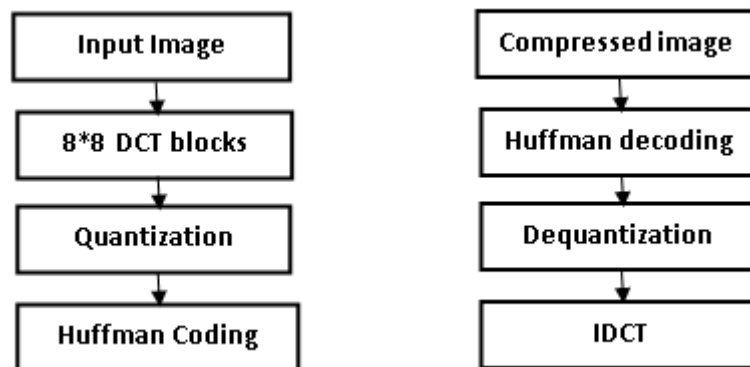
Figure 1. DCT-H, compression model (left) and decompression model (right).

Regarding the compression steps shown in Figure 1 (left), the input image is first loaded. Then, the loaded image is divided into blocks with 8*8 pixels. Forward 2D DCT function is applied on each block to obtain the DCT coefficients which are the 64 basis functions of the 8*8 pixel blocks from the original image. An example of 8*8 image luminance block, with a range of 8-bit values f(i, j) [0, 255], is illustrated in Figure 2 (left) [12].

Figure 2. An 8*8 block f(i, j) from an image (left) and its corresponding DCT coefficients F(u, v) (right).

As shown in Figure 2 (right), except the DC and the first few AC components representing low spatial frequencies, most of the DCT coefficients $F(u, v)$ have small magnitudes. This is because the pixel values in this block contain few high-spatial-frequency changes. We can see that the most information is accurately described by the first few components of the DCT coefficients. Therefore, the remaining components can be coarsely quantized, or even set to zero, with little signal distortion.

In the third step, quantization is applied on the DCT coefficients in order to minimize the number of output values to a high smaller set and discard the less important components. This example uses the luminance quantization table shown in Figure 3. The quantization step leads to high numbers of zeros repeated as shown in Figure 4. Therefore, in the last step of encoding, Huffman encoding can be applied to achieve better compression.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 3. The luminance quantization table.

-5	-4	9	-5	2	1	1	0
-11	-5	-2	0	1	0	0	-1
3	-6	4	0	-3	-1	0	1
0	1	-1	0	1	0	0	0
0	0	-1	0	0	1	0	0
0	-1	1	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 4. Quantization on DCT coefficients.

-80	-44	90	-80	48	40	51	0
-132	-60	-28	0	26	0	0	-55
42	-78	64	0	-120	-57	0	56
0	17	-22	0	51	0	0	0
0	0	-37	0	0	109	0	0
0	-35	55	-64	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 5. De-quantization results.

In our experiments, however, we use a uniform scalar quantizer with a scalar factor q = 20. This means that the DCT coefficients are divided by 20, which produces similar results to the results shown in Figure 4. See subsection "4.5 Experimental Setup". Contrariwise, for the decompression model steps shown in Figure 1 (right), the compressed image file is loaded and then Huffman decoding is applied. After that, dequantization is carried out to retain the DCT coefficients as shown in Figure 5. In the last step, inverse discrete cosine transformation (IDCT) is applied to reconstruct an approximated image of the original image.

Figure 6 shows the reconstructed 8*8 image block after applying IDCT. The figure shows that the reconstructed block is in close approximation of the original block, which generally cannot be visually noticed by the human eye in the image. To illustrate the quality of DCT compression, the error e = original 8*8 block (Figure 2, left) – reconstructed 8*8 block (Figure 6) is shown in Figure 7.

70	60	106	94	62	103	146	176
85	101	85	75	102	127	93	144
98	99	92	102	74	98	89	167
132	53	111	180	55	70	106	145
173	57	114	207	111	89	84	90
164	123	131	135	133	92	85	162
141	159	169	73	106	101	149	224
150	141	195	79	107	147	210	153

Figure 6. The reconstructed 8*8 image block.

0	-1	11	4	-15	27	-6	-31
2	-14	24	-23	-4	-11	-3	29
4	16	-24	20	24	1	11	-49
-12	13	-27	-7	-8	-18	12	23
-3	0	16	-2	-20	21	0	-1
5	-12	6	27	-3	2	14	-37
6	5	-6	-9	6	14	-47	44

Figure 7. Error of DCT compression.

4.3 Compression/Decompression Model (DWT-H)

Similar to DCT-H, Figure 8 portrays the framework used to implement the compression/decompression model for Discrete Wavelet Transformation combined with Huffman coding (DWT-H). The objective of the wavelet transform is to decompose the input signal into components some of which can be thresholded away. Moreover, the original image can be approximately reconstructed using these components. Two-level 2D DWT is applied on the input image using the MATLAB function "dwt2(input, 'haar')" to obtain the most significant components that occur in the upper left side of the image (LL). Figure 9 shows the two-level DWT of 256*256 image. LL band holds the approximated version of the original image and represents the general trend of pixel values of the input image. The other components can be efficiently encoded (after quantization) or even discarded, because they have little significant information.
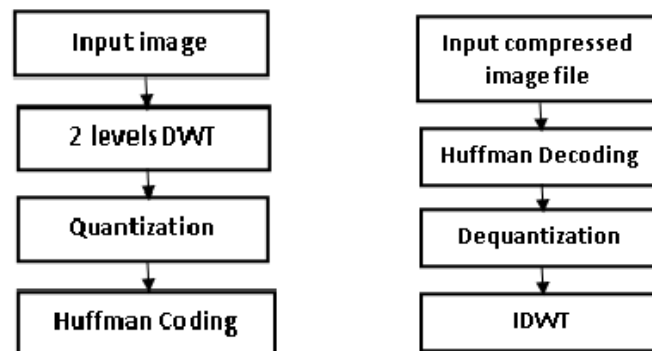
Figure 8. DWT-H, compression model (left) and decompression model (right).

After one level of 2D DWT is complete, the transformed image contains four subbands; LL, HL, LH and HH, standing for low-low, high-low and so on, as Figure 9 shows. The LL subband can be further decomposed to yield yet another level of decomposition (two-level DWT). This process can be continued until the desired number of decomposition levels is reached or the LL component only has a single element (left). Figure 10 shows an example of 16*16 gray image block (up) and its one-level 2D DWT (down). The two-level 2D DWT is also shown in Figure 11. We can see that, except the LL components, the other components have small values which can be coarsely quantized and coded with little signal distortion. In the third step of the compression model of DWT-H shown in Figure 8 (left), quantization is applied to the DWT components in order to minimize the number of output values. Finally, in the last step, Huffman encoding is applied for compression purpose. Contrariwise, for the decompression steps shown in Figure 8 (right), Huffman decoding is applied on the compressed image. Then, dequantization is carried out to retain the decoded DWT components. In the last step, inverse wavelet transformation (IDWT) is applied to reconstruct the image.

Figure 9. Two-level DWT transformation.

4.4 Evaluation Metrics

To evaluate the performance of both combined compression techniques; i.e., DCT-H and DWT-H,

"A Comparative Study of DCT and DWT Image Compression Techniques Combined with Huffman Coding , A. Maghari.

five performance metrics have been used; Compression Ratio (CR), Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) as well as compression time and decompression time. They are described as follows:
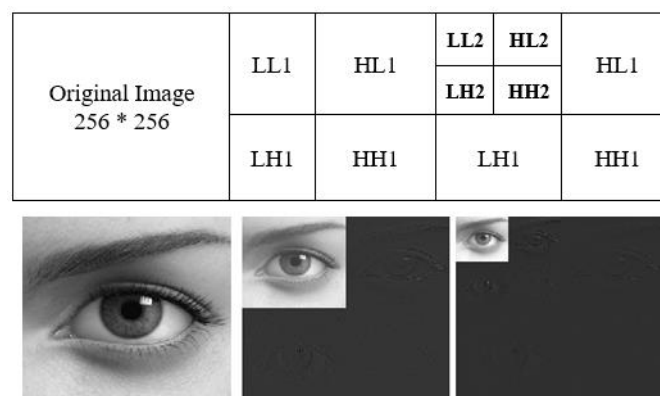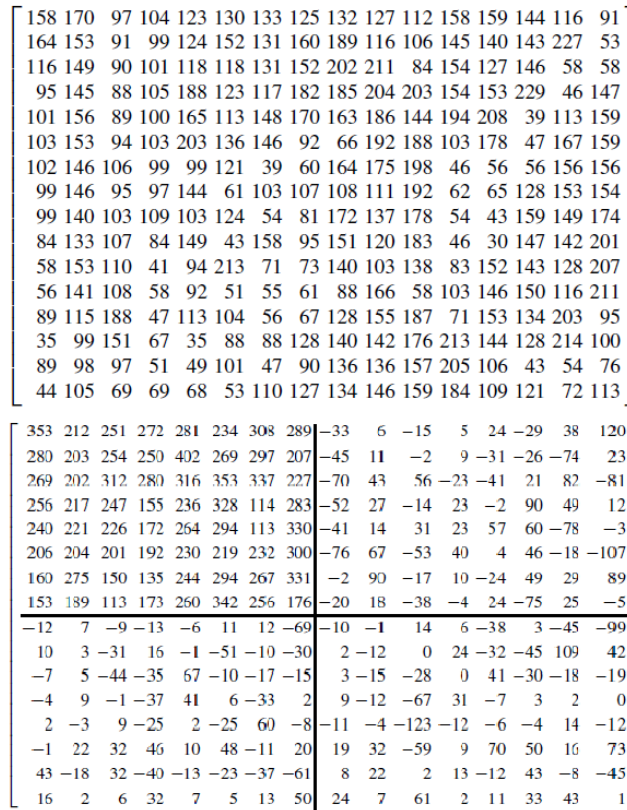
$$
\begin{bmatrix}
158 & 170 & 97 & 104 & 123 & 130 & 133 & 125 & 132 & 127 & 112 & 158 & 159 & 144 & 116 & 91 \\
164 & 153 & 91 & 99 & 124 & 152 & 131 & 160 & 189 & 116 & 106 & 145 & 140 & 143 & 227 & 53 \\
116 & 149 & 90 & 101 & 118 & 118 & 131 & 152 & 202 & 211 & 84 & 154 & 127 & 146 & 58 & 58 \\
95 & 145 & 88 & 105 & 188 & 123 & 117 & 182 & 185 & 204 & 203 & 154 & 153 & 229 & 46 & 147 \\
101 & 156 & 89 & 100 & 165 & 113 & 148 & 170 & 163 & 186 & 144 & 194 & 208 & 39 & 113 & 159 \\
103 & 153 & 94 & 103 & 203 & 136 & 146 & 92 & 66 & 192 & 188 & 103 & 178 & 47 & 167 & 159 \\
102 & 146 & 106 & 99 & 99 & 121 & 39 & 60 & 164 & 175 & 198 & 46 & 56 & 56 & 156 & 156 \\
99 & 146 & 95 & 97 & 144 & 61 & 103 & 107 & 108 & 111 & 192 & 62 & 65 & 128 & 153 & 154 \\
99 & 140 & 103 & 109 & 103 & 124 & 54 & 81 & 172 & 137 & 178 & 54 & 43 & 159 & 149 & 174 \\
84 & 133 & 107 & 84 & 149 & 43 & 158 & 95 & 151 & 120 & 183 & 46 & 30 & 147 & 142 & 201 \\
58 & 153 & 110 & 41 & 94 & 213 & 71 & 73 & 140 & 103 & 138 & 83 & 152 & 143 & 128 & 207 \\
56 & 141 & 108 & 58 & 92 & 51 & 55 & 61 & 88 & 166 & 58 & 103 & 146 & 150 & 116 & 211 \\
89 & 115 & 188 & 47 & 113 & 104 & 56 & 67 & 128 & 155 & 187 & 71 & 153 & 134 & 203 & 95 \\
35 & 99 & 151 & 67 & 35 & 88 & 88 & 128 & 140 & 142 & 176 & 213 & 144 & 128 & 214 & 100 \\
89 & 98 & 97 & 51 & 49 & 101 & 47 & 90 & 136 & 136 & 157 & 205 & 106 & 43 & 54 & 76 \\
44 & 105 & 69 & 69 & 68 & 53 & 110 & 127 & 134 & 146 & 159 & 184 & 109 & 121 & 72 & 113
\end{bmatrix}
$$

$$
\begin{bmatrix}
353 & 212 & 251 & 272 & 281 & 234 & 308 & 289 & -33 & 6 & -15 & 5 & 24 & -29 & 38 & 120 \\
280 & 203 & 254 & 250 & 402 & 269 & 297 & 207 & -45 & 11 & -2 & 9 & -31 & -26 & -74 & 23 \\
269 & 202 & 312 & 280 & 316 & 353 & 337 & 227 & -70 & 43 & 56 & -23 & -41 & 21 & 82 & -81 \\
256 & 217 & 247 & 155 & 236 & 328 & 114 & 283 & -52 & 27 & -14 & 23 & -2 & 90 & 49 & 12 \\
240 & 221 & 226 & 172 & 264 & 294 & 113 & 330 & -41 & 14 & 31 & 23 & 57 & 60 & -78 & -3 \\
206 & 204 & 201 & 192 & 230 & 219 & 232 & 300 & -76 & 67 & -53 & 40 & 4 & 46 & -18 & -107 \\
160 & 275 & 150 & 135 & 244 & 294 & 267 & 331 & -2 & 90 & -17 & 10 & -24 & 49 & 29 & 89 \\
153 & 189 & 113 & 173 & 260 & 342 & 256 & 176 & -20 & 18 & -38 & -4 & 24 & -75 & 25 & -5 \\
-12 & 7 & -9 & -13 & -6 & 11 & 12 & -69 & -10 & -1 & 14 & 6 & -38 & 3 & -45 & -99 \\
10 & 3 & -31 & 16 & -1 & -51 & -10 & -30 & 2 & -12 & 0 & 24 & -32 & -45 & 109 & 42 \\
-7 & 5 & -44 & -35 & 67 & -10 & -17 & -15 & 3 & -15 & -28 & 0 & 41 & -30 & -18 & -19 \\
-4 & 9 & -1 & -37 & 41 & 6 & -33 & 2 & 9 & -12 & -67 & 31 & -7 & 3 & 2 & 0 \\
2 & -3 & 9 & -25 & 2 & -25 & 60 & -8 & -11 & -4 & -123 & -12 & -6 & -4 & 14 & -12 \\
-1 & 22 & 32 & 46 & 10 & 48 & -11 & 20 & 19 & 32 & -59 & 9 & 70 & 50 & 16 & 73 \\
43 & -18 & 32 & -40 & -13 & -23 & -37 & -61 & 8 & 22 & 2 & 13 & -12 & 43 & -8 & -45 \\
16 & 2 & 6 & 32 & 7 & 5 & 13 & 50 & 24 & 7 & 61 & 2 & 11 & 33 & 43 & 1
\end{bmatrix}
$$

Figure 10. A 16*16 block from a gray image (up) and its one-level 2D DWT (down).

$$
\begin{bmatrix}
558 & 451 & 608 & 532 & 75 & 26 & 94 & 25 & -33 & 6 & -15 & 5 & 24 & -29 & 38 & 120 \\
463 & 511 & 627 & 566 & 66 & 68 & -43 & 68 & -45 & 11 & -2 & 9 & -31 & -26 & -74 & 23 \\
464 & 401 & 478 & 416 & 14 & 84 & -97 & -229 & -70 & 43 & 56 & -23 & -41 & 21 & 82 & -81 \\
422 & 335 & 477 & 553 & -88 & 46 & -31 & -6 & -52 & 27 & -14 & 23 & -2 & 90 & 49 & 12 \\
14 & 33 & -56 & 42 & 22 & -43 & -36 & 1 & -41 & 14 & 31 & 23 & 57 & 60 & -78 & -3 \\
-13 & 36 & 54 & 52 & 12 & -21 & 51 & 70 & -76 & 67 & -53 & 40 & 4 & 46 & -18 & -107 \\
25 & -20 & 25 & -7 & -35 & 35 & -56 & -55 & -2 & 90 & -17 & 10 & -24 & 49 & 29 & 89 \\
46 & 37 & -51 & 51 & -44 & 26 & 39 & -74 & -20 & 18 & -38 & -4 & 24 & -75 & 25 & -5 \\
-12 & 7 & -9 & -13 & -6 & 11 & 12 & -69 & -10 & -1 & 14 & 6 & -38 & 3 & -45 & -99 \\
10 & 3 & -31 & 16 & -1 & -51 & -10 & -30 & 2 & -12 & 0 & 24 & -32 & -45 & 109 & 42 \\
-7 & 5 & -44 & -35 & 67 & -10 & -17 & -15 & 3 & -15 & -28 & 0 & 41 & -30 & -18 & -19 \\
-4 & 9 & -1 & -37 & 41 & 6 & -33 & 2 & 9 & -12 & -67 & 31 & -7 & 3 & 2 & 0 \\
2 & -3 & 9 & -25 & 2 & -25 & 60 & -8 & -11 & -4 & -123 & -12 & -6 & -4 & 14 & -12 \\
-1 & 22 & 32 & 46 & 10 & 48 & -11 & 20 & 19 & 32 & -59 & 9 & 70 & 50 & 16 & 73 \\
43 & -18 & 32 & -40 & -13 & -23 & -37 & -61 & 8 & 22 & 2 & 13 & -12 & 43 & -8 & -45 \\
16 & 2 & -6 & -32 & -7 & 5 & -13 & -50 & 24 & 7 & -61 & 2 & 11 & -33 & 43 & 1
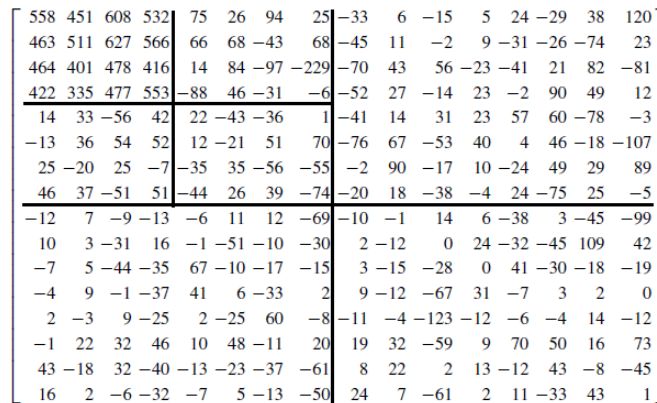\end{bmatrix}
$$

Figure 11. Two-level 2D DWT of the 16*16 block shown in Figure 10.

4.4.1 Compression Ratio

The Compression Ratio (CR) is defined as the ratio of the number of bytes of the original image to that of the compressed image. It can also be described as the ratio of the size of the original image to the size of the compressed image. The formula of CR is given in Equation () [19]. The higher the ratio is, the better is the compression technique.

$$CR = \frac{Number\ of\ bytes\ of\ original\ image}{Number\ of\ bytes\ of\ compresses\ image} \tag{7}$$

4.4.2 Peak Signal-to-Noise Ratio (PSNR)

PSNR is a measure factor for the quality of the compressed image to the original image. It determines

81

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

the error between the original and the reconstructed image. A high PSNR value means a high-quality image. The objective of this factor is to measure partially the human visual response to image quality[20]. The mathematical expression of PSNR is given in Equation () [18].

$$PSNR = 10 \, \log_{10} \left(\frac{255^2}{MSE} \right)$$ (8)

where:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \| I_0(i,j) - I_r(i,j) \|^2$$

m,n = image size > 0, I$_0$ = original value and I$_r$ = compressed value.

The Mean Square Error (MSE) [19] is the cumulative squared error between the original image and the compressed image. A low value of MSE means a higher value of PSNR. A high PSNR provides a better image quality after the reconstruction of the image.

4.4.3 Structural Similarity Index Measure (SSIM)

SSIM is a well-known quality metric used to measure the structural distortion between two images. It was developed by Wang et al. [21] and is recently used by many researchers [22], [23], [24] for image quality assessment. As opposed to SSIM, some studies revealed that PSNR performs badly in discriminating structural content in images, since various types of degradations applied to the same image can yield the same value of PSNR [25]. Generally, the value of PSNR can be predicted from SSIM and *vice-versa* [26]. SSIM is designed to compare between the original and the distorted images (x, y) using three factors: luminance, contrast and structural factors. It is defined as:

$$SSIM\,(x,y) \; = \; [l(x,y)]^\alpha . [c(x,y)]^\beta . [s(x,y)]^\gamma$$ (9)

where, $l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}$, $c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$ and $s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$

where $\mu_x, \mu_y, \sigma_x, \sigma_y, and\ \sigma_{xy}$ are the local means, standard deviations and cross-covariance for images x and y.

The first term *l(x, y)* is the luminance comparison function, which measures the closeness of the two images' mean luminance ($\mu_x\ and\ \mu_y$). The second term *c(x, y)* is the contrast comparison function, which measures the closeness of the contrast of the two images. Here, the contrast is measured by the standard deviation $\sigma_{x\ and}\ \sigma_y$. The third term *s(x, y)* is the structure comparison function, which measures the correlation coefficient between the two images *x* and *y*. Note that σ_{xy} is the covariance between *x* and *y*. The positive values of the SSIM index are in the range [0-1]. A value of 0 means no correlation between images and 1 means *x = y*. The parameters α, β and γ indicate the importance of the three components, which are set to 1. C_1, C_2 and C_3 are constants and $C_3 = C_2/2$. So, the SSIM simplifies to:

$$SSIM(x,y) \; = \; \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$ (10)

4.5 Experimental Setup

We have used MATLAB to conduct the experiments on a CPU @ 2.50GHz with an Intel(R) Core

(TM) i5-2450M processor. There are different images used for experimentation. We choose as examples five different BMP gray images to conduct our experiments. BMP (BitMap) is standard file format for Microsoft Windows, which can be stored uncompressed. The five gray images have the same size of 256*256 = 65536 bytes (66 KB). The size of 256×256 pixels is chosen to ease implementing DCT and WDT, where the image is split into 8x8 blocks, which are then transformed into the frequency plane using Fast Discrete Cosine Transform (FDCT). For DWT, the 2D input image of size 256 x 256 is subjected to two-level decomposition using discrete Haar wavelet transform function. The scalar factor (q) for uniform quantization is set to 10, which affects the quantization steps for Huffman coding. Quantization is the main source of losing information. So, we choose a

small value for q to obtain a small MSE. For DCT, a block image size of 8×8 is used with a scalar factor q = 20. There are no precise rules for selecting the scalar factor q values. The trade-off between quality and CR can be controlled by the scalar factor q which will define the size of the frequency components [18]. Different scalar values (10 and 20) were chosen for DWT and DCT during the evaluation process to have suitable compression ratio and MSE. To determine the execution time for compression and decompression processes, we use tic and toc MATLAB functions together in order to measure the amount of time MATLAB takes and display the time in seconds.

5. RESULTS AND DISCUSSION

5.1 Results

Figure 12 shows the test images and their corresponding resulting compressed images using DWT-H and DCT-H. The experimental results with the comparison factors have been arranged in Table 1. Table 1 shows the results of the five comparison factors obtained by applying DCT and Huffman coding. It shows the compression ratio (CR), PSNR, MSE, SSIM, compression time and decompression time. Table 2 shows the same five factors for DWT combined with Huffman coding (DWT-H). Regarding the performance metrics, the results show that DWT-H outperforms DCT-H in that DWT-H produced mostly higher CR, higher PSNR, higher SSIM and lower MSE than DCT-H, as graphically shown in Figure 13. Performance metric values obtained are as shown in Table 1 and 2.

In contrast, for the compression/decompression time, the results show that DCT-H is less time-consuming than DWT-H. The compression/decompression time values obtained are shown graphically in Figure 14.

Table 1. The comparison factor results of DCT-H.

Image name	CR	PSNR	MSE	SSIM	Decode Time (s)	Code Time (s)
Camera man	1.72	36.04	16.3	0.575	0.12	1.14
Rice	2.56	38.04	10.28	0.772	0.12	0.2
MRI	3.37	39.37	7.57	0.898	0.11	0.15
Eye	2.66	36.7	13.9	0.734	0.14	0.16
Tiger	1.5	35.6	17.9	0.822	0.12	0.14
Average results	2.36	37.15	13.19	0.76	0.122	0.358

Table 2. The comparison factor results of DWT-H.

Image name	CR	PSNR	MSE	SSIM	Decode Time	Code Time
Camera man	1.96	40.8	5.4	0.712	2.44	2.7
Rice	1.97	41.7	4.4	0.856	2.4	2.7
MRI	8.24	49.6	0.72	0.993	0.65	0.7
Eye	2.1	40.1	6.4	0.854	2.27	2.64
tiger	1.6	40.5	5.8	0.851	2.9	3.17
Average results	3.17	42.5	4.54	0.85	2.13	2.38

5.2 DISCUSSION

In this study and due to our comparison study, we applied DCT and DWT algorithms in combination with Huffman algorithm. For the DCT transform, the image is split into 8x8 blocks which are then transformed into the frequency plane using Fast Discrete Cosine Transform (FDCT). After the transform, most of information is concentrated to a few low-frequency components. These components are then quantized in order to reduce the number of bits needed to represent the image. This quantization step will lower the quality of the image by reducing the precision of the components. Regarding the DWT algorithm, as a result of applying DWT on the input image, it is divided into four non-overlapping multi-resolution sub-bands; LL, LH, HL and HH. The sub-band LL represents the coarse-scale DWT coefficients while the sub-bands LH, HL and HH represent the fine-scale DWT coefficients. Then, sub-band quantization and coding are used to reduce the number of bits needed to represent the image. This step will lower the quality of the image but not like DCT, because in DCT, the information is concentrated on fewer components that can be affected more sensitively by
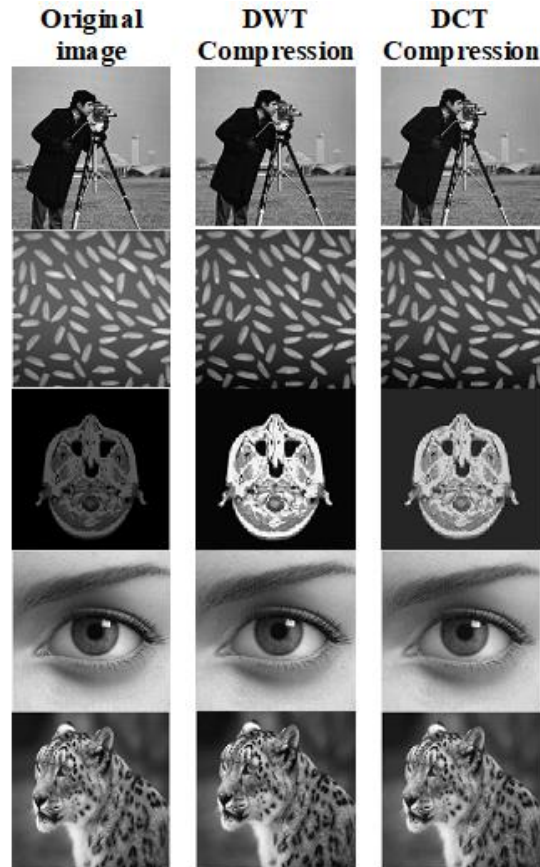
Figure 12. The five test images (Camera man, Rice, MRI, Eye, Tiger) and the corresponding compressed images using DCT-H and DWT-H.
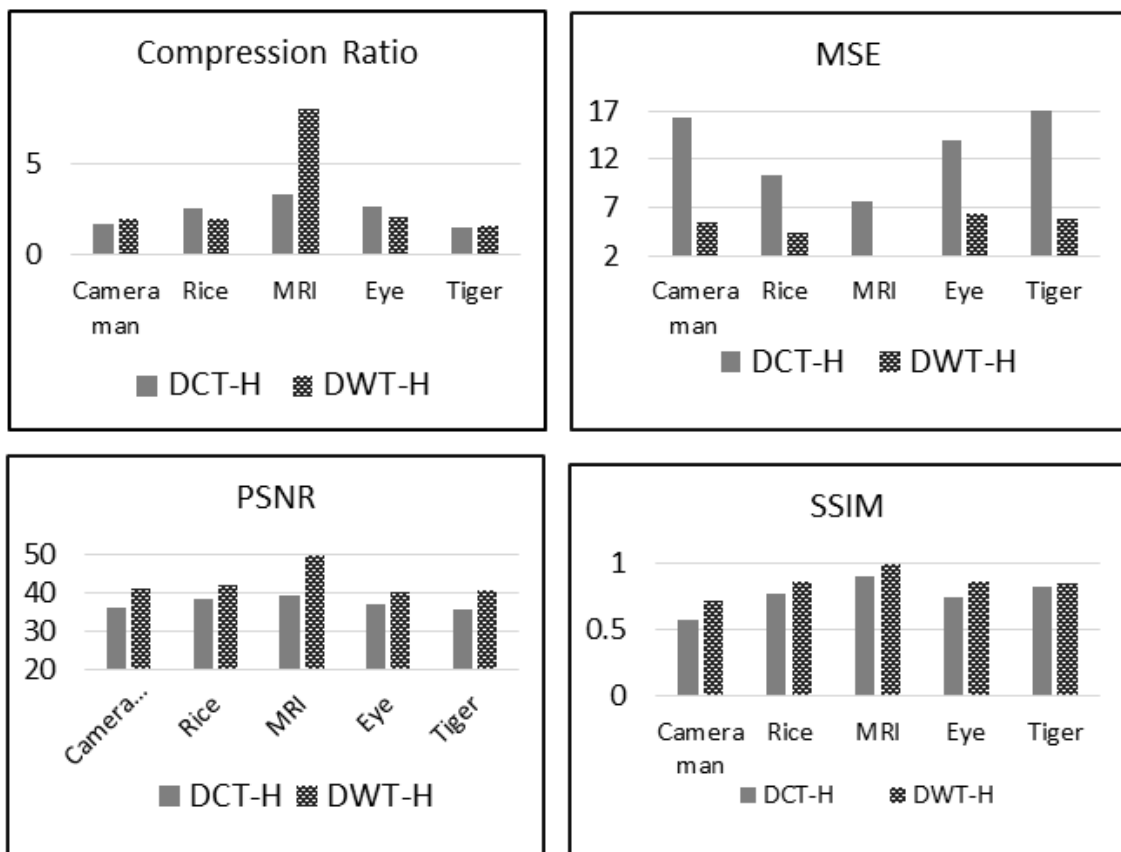

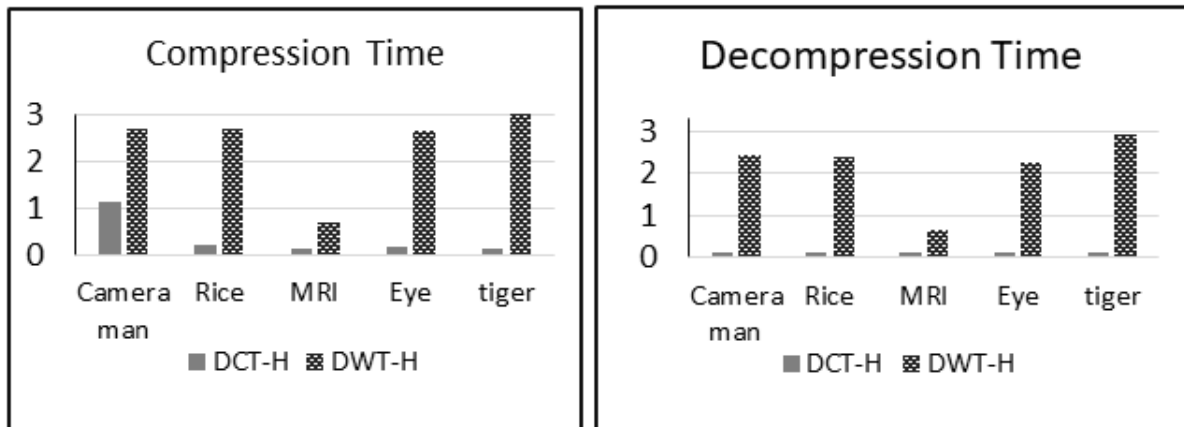Figure 13. DCT-H and DWT-H: Compression Ratio, MSE, PSNR and SSIM.

Figure 14: Compression and decompression time of DCT-H and DWT-H.

quantization than in DWT. Moreover, DWT avoids blocking artifacts that can happen by dividing the input image into blocks as in DCT [9].

The results illustrate that DWT combined with Huffman (DWH-H) coding give mostly higher compression ratio, higher PSNR and higher SSIM than DCT combined with Huffman coding (DCT-H). At the same time, we found that compression and decompression time of DCT-H was less than in DWT-H. Our appreciation for the appearance of these results is based on the extent of similarity in output values, when DCT or DWT algorithms are applied on the test images. This means that when DCT or DWT output coefficients are mostly very small, then the quantization does not cause big loss in the information. The quantization is used to reduce the number of distinct output values to a much smaller set to efficiently represent the source output using any coding techniques such as Huffman coding [12]. Using a large scalar value for quantization gives high compression ratio and low-quality image reconstruction. So, in our experiments, we used a constant small scalar value (q) for quantization (q=10 for DWT, q=20 for DCT) in order to obtain small MSE and high-quality reconstruction. According to our observations, we conclude that using DWT with Huffman algorithm achieves better compression ratio, PSNR and SSIM than using DCT with Huffman algorithm. Hence, performance of DWT combined with Huffman coding is comparatively higher than that of DCT combined with Huffman coding. Our results are consistent with [8], where DWT algorithm performed much better than DCT algorithm in terms of Compression Ratio and Peak Signal-to-Noise Ratio (PNSR). On the other hand, DCT-H is less time-consuming than WDT-H.

6. CONCLUSION

In this paper, we performed a comparative study based on DCT and DWT in combination with Huffman coding in terms of Compression Ratio (CR), Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) as well as compression and decompression time. Our experiments were conducted on five different BMP file images; Camera man, Rice, MRI, Eye and Tiger. The experimental results showed that using DWT with Huffman algorithm (DWT-H) achieves a comparable compression ratio with higher PSNR and less MSE than DCT with Huffman algorithm (DCT-H). At the same time, we found out that DWT-H is time-consuming in terms of compression and decompression. The average CR metric value of the five images was 2.36 for DCT-H and 3.17 for DWT-H. Regarding MSE, PSNR and SSIM, the average result values of DCT-H were: MSE = 13.19, PSNR = 37.15 and SSIM = 0.76. WDT-H has the average result values of MSE = 4.54, PSNR = 42.5 and SSIM = 0.85. On the other hand, DCT-H outperforms DWT-H in terms of compression and decompression execution times. DCT-H has an average execution time of 0.358s for compression and an average execution time 0.122s for decompression, while WDT-H has 2.38s compression time and 2.13s decompression time. In future works, we will extend our experimental study to other image compression algorithms and use different coding techniques.

REFERENCES

[1]   M. Gupta and A. K. Garg, "Analysis of Image Compression Algorithm Using DCT," Int. J. Eng. Res. Appl., vol. 2, no. 1, pp. 515–521, 2012.

[2]   T. Sheltami, M. Musaddiq and E. Shakshuki, "Data Compression Techniques in Wireless Sensor Networks," Futur. Gener. Comput. Syst., vol. 64, pp. 151–162, 2016.

[3]   J. Uthayakumar, T. Vengattaraman and P. Dhavachelvan, "A Survey on Data Compression Techniques: From the Perspective of Data Quality, Coding Schemes, Data Type and Applications," J. King Saud Univ. Inf. Sci., 2018.

[4]   A. Mr, S. Subramanian and S. Mr, "Comparing PSNR of Different Image Transforms (DCT, DFT, DWT, DHT, DTT)," 2018.

[5]   Nashar Luthfi Sugara, T. W. Purboyo and A. L. Prasasti, "Comparative Analysis of Image Compression Using Huffman and DCT Method on JPG Image," J. Eng. Appl. Sci., vol. 13, pp. 4447-4452, 2018.

[6]   G. Suseela and Y. A. V. Phamila, "Energy Efficient Image Coding Techniques for Low Power Sensor Nodes: A Review," Ain Shams Eng. J., vol. 9, no. 4, pp. 2961–2972, 2018.

[7]   W. Z. Wahba and A. Y. A. Maghari, "Lossless Image Compression Techniques: Comparative Study," Int. Res. J. Eng. Technol., vol. 3, no. 2, pp. 1–9, 2016.

[8]   A. H. M. J. I. Barbhuiya, T. A. Laskar and K. Hemachandran, "An Approach for Color Image Compression of JPEG and PNG Images Using DCT and DWT," Proc. of the International Conference on Computational Intelligence and Communication Networks (CICN), pp. 129–133, 2014.

[9]   N. Saroya and P. Kaur, "Analysis of Image Compression Algorithm Using DCT and DWT Transforms," Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 4, no. 2, 2014.

[10]   T. Kumar and R. Kumar, "Medical Image Compression Using Hybrid Techniques of DWT, DCT and Huffman Coding," Int. J. Innov. Res. Electr. Electron. Instrum. Control Eng., vol. 3, no. 2, pp. 54–60, 2015.

[11]   G. K. Wallace, "The JPEG Still Picture Compression Standard," IEEE Trans. Consum. Electron., vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[12]   Ze-Nian Li, Drew, Jiangchuan Liu and Mark S. Drew, Fundamentals of Multimedia, 2nd Ed., Springer International Publishing, 2014.

[13]   C. Wang, R. Xiong, H. He, Y. Zhang and W. Shen, "Comparison of Decomposition Levels for Wavelet Transform-based Energy Management in a Plug-in Hybrid Electric Vehicle," Journal of Cleaner Production, vol. 210, pp. 1085–1097, 2019.

[14]   A. J. Maan, "Analysis and Comparison of Algorithms for Lossless Data Compression," Int. J. Inf. Comput. Technol., vol. 3, no. 3, pp. 139–146, 2013.

[15]   A. M. G. Hnesh and H. Demirel, "DWT-DCT-SVD Based Hybrid Lossy Image Compression Technique," Proc. of the International Conference on Image Processing, Applications and Systems (IPAS), pp. 1–5, 2016.

[16]   R. Kher and Y. Patel, "Medical Image Compression Framework Based on Compressive Sensing, DCT and DWT," Biology, Engineering and Medicine, vol. 2, no. 2, pp. 1–4, 2017.

[17]   S. Sharma and S. Kaur, "Image Compression Using Hybrid of DWT, DCT and Huffman Coding," Int. J. Sci. Emerg. Technol. with Latest Trends, vol. 5, no. 1, pp. 19–23, 2013.

[18]   O. Ghorbel, "DCT & DWT Image Compression Algorithms in Wireless Sensor Networks: Comparative Study and Performance Analysis," Int. J. Wirel. Mob. Networks, 2013.

[19]   A. Katharotiya, S. Patel and M. Goyani, "Comparative Analysis between DCT and DWT Techniques of Image Compression," J. Inf. Eng. Appl., vol. 1, no. 2, pp. 9–17, 2011.

[20]   R. Monika, S. Dhanalakshmi and S. Sreejith, "Coefficient Random Permutation Based Compressed Sensing for Medical Image Compression," Advances in Electronics, Communication and Computing, Springer, pp. 529–536, 2018.

[21]   Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli et al., "Image Quality Assessment: From Error Visibility to Structural Similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, 2004.

"A Comparative Study of DCT and DWT Image Compression Techniques Combined with Huffman Coding , A. Maghari.

[22]   R. Reisenhofer, S. Bosse, G. Kutyniok and T. Wiegand, "A Haar Wavelet-based Perceptual Similarity Index for Image Quality Assessment," Signal Processing: Image Communication, vol. 61, pp. 33–43, 2018.

[23]   K. Ma, Z. Duanmu, H. Yeganeh and Z. Wang, "Multi-exposure Image Fusion by Optimizing a Structural Similarity Index," IEEE Transactions on Comput. Imaging, vol. 4, no. 1, pp. 60–72, 2018.

[24]   S. Pistonesi, J. Martinez, S. M. Ojeda and R. Vallejos, "Structural Similarity Metrics for Quality Image Fusion Assessment: Algorithms," Image Process. Line, vol. 8, pp. 345–368, 2018.

[25]   Z. Wang and A. C. Bovik, "Mean Squared Error: Love It or Leave It? A New Look at Signal Fidelity Measures," IEEE Signal Processing Magazine, vol. 26, no. 1, pp. 98–117, 2009.

[26]   A. Hore and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," Proc. of the 20th International Conference on Pattern Recognition, pp. 2366–2369, 2010.

ملخص البحث:

لقـد شـاع اسـتخدام تقنيـات ضـغط الصـور لتخـزين البيانـات ونقلهـا، الأمـر الـذي يتطلـب حيـزاً تخزينيـاً كبيـراً وسـرعة نقـل عاليـة. ويـؤدي النمـو السـريع للصـور عاليـة الجـودة إلـى تنـامي الطلـب علـى تقنيـات فعالـة لتخـزين البيانـات وتبادلهـا عبـر الإنترنـت. فـي هـذه الورقـة، نقـدم دراسـة مقارنـة بـين خوارزميـات تقنيتـي DCT و DWT مـع اسـتخدام ترميـز هوفمـان. والمقارنـة فـي هـذه الدراسـة مبنيـة علـى خمسـة عوامـل هـي: معـدل الضـغط (CR)، ومتوسـط مربـع الخطـأ (MSE)، وأعلـى نسـب الإشـارة إلـى الضـجيج (PSNR)، ومقياس عامل التشابه البنيوي (SSIM)، وزمن الضغط/إزالة الضغط.

أجريـت التجـارب علـى خمـس صـور تنتمـي إلـى ملفـات رماديـة التـدريج. وبينـت النتـائج أن الترميـز باسـتخدام تقنيـة DWT يضـاهي مثيلـه باسـتخدام تقنيـة DCT مـن حيـث معـدل الضـغط (CR)، ويتقـدم عليـه فيمـا يتعلـق بمتوسـط مربـع الخطـأ وأعلـى نسـب الإشـارة إلـى الضـجيج ومقيـاس عامـل التشـابه البنيـوي. وكانـت نتـائج الصـور الخمـس المفحوصـة فيمـا يخـصّ معـدل الضـغط فـي المعـدل 2.36 و 3.17 علـى الترتيـب لكـل مـن تقنيـة DCT وتقنيـة DWT. عـلاوة علـى ذلـك، كـان معـدل النتـائج لتقنيـة DCT كمـايلي: MSE=13.19؛ PSNR= 37.15؛ SSIM= 0.76. وعلـى الجانـب الآخـر، كـان معـدل النتـائج لتقنيـة DWT علـى النحـو الآتـي: MSE= 4.54؛ PSNR= 42.5؛ SSIM= 0.85.

مـن جانـب آخـر، تتفـوق تقنيـة DCT علـى تقنيـة DWT مـن حيـث زمـن تنفيـذ الضـغط وإزالـة الضـغط، بحيـث كـان معـدل النتـائج لتقنيـة DCT 0.358 ثانيـة لـزمن الضـغط و 0.122 ثانيـة لـزمن إزالـة الضـغط؛ فـي حـين كـان زمـن الضـغط وزمـن إزالـة الضـغط 2.38 ثانية و 2.13 ثانية على الترتيب لتقنية DWT.

87

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

A SURVEY ON AGE-INVARIANT FACE RECOGNITION METHODS

Zahra Mortezaie[1] and Hamid Hassanpour[2]

ABSTRACT

Face recognition is used in many security and surveillance applications. Some issues, such as aging, partial occlusion, variation in pose and illumination and facial expression directly affect the performance of face recognition approaches. Usually, in many applications, such as checking the passport and visa, images in the database are not updated continuously. In these cases, aging leads to change the important features of the face image. Hence, face recognition across aging can be considered as a common issue in many security and surveillance systems. In this paper, some existing face recognition approaches, in terms of robustness to aging, have been reviewed briefly. Also, the experimental results from these methods have been compared using the common databases in age-invariant face recognition applications. The comparison results indicate that the approaches which consider both component-based representation of facial images and identity factors outperform the other existing methods.

KEYWORDS

Face recognition, Aging, Age-invariant, Facial image.

1. INTRODUCTION

Face recognition is considered as an effective and applicable identification and verification technique, as it is an inexpensive and non-intrusive approach [1]. It has been used in many applications, such as security systems, card verification, video surveillance, credit criminal identification, person identification [2], passport renewal, law enforcement and biometric authentication [3]. There are three stages in face recognition systems [1]. In face recognition, the facial image is detected firstly. Then, suitable features are extracted from the image. Finally, the extracted features are compared with those from the database using a similarity measure.

Accuracy of face recognition methods directly depends on the used facial image. Indeed, some variations in facial images, such as age, pose, illumination, partial occlusion and facial expressions, reduce the accuracy of face recognition methods [3]. Among these variations, aging leads to change the important features of facial images, such as texture and shape. In many security and surveillance applications, the available databases are not updated during the time. Also, some existing non-age-invariant methods may correctly recognize the captured facial images using images captured within a one-year interval. In the case of facial images captured within a longer period of time, many changes may appear in comparison to previously captured images, especially in cases in which the images include a person's face before and after the age of puberty. Hence, it is necessary to consider the effect of aging in face recognition techniques [3]-[6].

There is some research on face recognition which considers age progression. This research can be categorized in three groups: age estimation approaches [7]-[20], aging simulation approaches [21]-[26] and age-invariant face recognition approaches [27]-[29], [31], [34]-[39], [41]-[47], [49]-[50]. In the first group, the age is estimated from the person's facial image. In the second group, a computational model is provided for modeling the facial appearances across aging. In the third group, age-invariant features are extracted for robust face recognition. Generally, age-invariant face recognition methods can be divided into two groups: generative [27]-[29], [31] and non-generative methods [34]-[39], [41]-[44]. In the former group, aging is initially simulated using a computational model. Then, the perceived image of the subject is normalized to eliminate aging variations. But, age-invariant features are extracted from the facial images in the latter group [4]. Recently, new methods, such as deep neural networks, have been used in age-invariant face recognition approaches [45]-[47], [49]-[50].

Z. Mortezaie and H. Hassanpour are with Computer Engineering Department, Shahrood University of Technology, Shahrood, Iran. Emails: [1]zm.mortezaie@gmail.com and [2]h hassanpour@yahoo.com

Deep neural networks, such as auto-encoder neural networks, can both extract features and model the face across aging. Hence, these methods can be categorized in both generative and non-generative face recognition methods across aging. Therefore, we can consider a new category in addition to generative and non-generative methods, named deep neural networks.

This paper aimed to review and compare the performance of the existing age-invariant face recognition methods. The rest of this paper is organized as follows. The age-invariant face recognition methods are reviewed in Section 2. The common existing databases for age-invariant face recognition approaches are introduced in Section 3. The experimental results from the reviewed methods are compared in Section 4. Finally, the conclusions are driven in Section 5.

2. AGE-INVARIANT FACE RECOGNITION

As mentioned earlier, age-invariant face recognition methods are generative or non-generative approaches and those which use deep neural networks. In this section, some existing age-invariant face recognition methods are reviewed briefly.

2.1 Generative Methods

In [27], a computational model has been proposed to describe the changes of shape and texture in the facial images across aging. In this method, a muscle-based geometrical change model has been proposed to model the aging progression in adulthood. In this model, the changes of physical properties and geometric orientations of the facial muscles have been described throughout adulthood. Also, in this method, the facial wrinkles and other skin traits that appeared across aging were modeled by proposing the image gradient-based texture transformation function.

In [28]-[29], a 2D/3D face aging pattern space has been provided to synthesize a facial image to match the target face image before recognition. In [28], at first, the facial aging is simulated using a 2D face aging model. Then, the aging model is used for face across aging. This model is proposed based on non-negative matrix factorization (NMF) [30] with sparseness constraints.

In the method introduced in [31], the recognition is performed in two stages. At first, a maximum *a posteriori* solution is used based on principal component analysis (PCA) factorization in order to decrease the search space. Then, a graph matching approach is used to investigate the matching between the probe image and the gallery image. Indeed, in [31], the facial images are presented using graph-based features. In this method, the aging of each face image is modeled using Gaussian mixture model (GMM). This model has considered the shape and texture information variations for each face image across aging. In this method, the feature points are extracted using a modified local feature analysis [32]. The modified local feature analysis uses the Fisher score [33] to extract the feature points. The feature descriptor is obtained *via* applying the uniform local binary pattern (LBP) operator on the feature point.

As the aging is a complex process, generative methods are not usually able to create the face model representing the aging process. Also, the limitation on the number of training data increases this problem. Besides, in these methods, extra information, such as accurate age labels for the training data and the landmark point locations for each face image, are required to create the face aging model. Also, in these methods, the used face images should be captured in ideal conditions, such as frontal pose, normal illumination and expression [5]. So, generative methods do not work well in real-world face recognition. Hence, in the age-invariant face recognition approaches, non-generative methods have been extended more than generative methods.

2.2 Non-Generative Methods

In [34], an age-invariant face recognition method was proposed based on the Eigen space techniques and the Bayesian model. In this method, a Bayesian age-difference classifier was built on a probabilistic Eigen space framework. This classifier is used to model the differences between face images across aging for individuals as well as to model the differences between various persons face images.

The gradient orientation information is a more robust feature in comparison with the other appearance

features in changing the age. Hence, in [35]-[36], the gradient orientation pyramid (GOP) feature was used to model the differences between person faces. In this method, the GOP feature was combined with the support vector machine (SVM) for face verification.

In [37], a discriminative model has been proposed considering the scale-invariant feature transform (SIFT) and multi-scale local binary patterns (MLBPs) as the local features for each face. The SIFT-based local features and MLBP-based local features have a high-dimensional feature space. Hence, in this method, an algorithm named multi-feature discriminant analysis (MFDA) was developed to reduce dimensionality. In MFDA, local descriptors were combined to create a robust decision rule by a random subspace fusion model.

The number of the considered face features can be an effective parameter in extracting the appropriate discriminative information. Hence, in [38], a multi-view discriminative learning (MDL) method was proposed for age-invariant face recognition. In this method, the SIFT, LBP and GOP descriptors have been used to extract the discriminative information from the facial images. Then, a discriminative learning method with multi-view feature representations; namely, MDL, was developed. Indeed, the MDL has been used to simultaneously minimize the variations in each feature class, maximize the variations between the feature classes and maximize the correlation of the feature classes from the same person.

Age variation affects each component of the face differently. Hence, in [39], a component-based method has been proposed for age-invariant face recognition. In this method, the components of the face are determined automatically using an active shape model (ASM) [40]. Then, the MLBP and SIFT features for each component are used in a random subspace linear discriminant analysis (LDA) for classification.

In [41], at first some pre-processing tasks, such as pose correction, illumination and periocular region normalization, are applied on the facial images. Then, the Walsh-Hadamard transform encoded local binary patterns (WLBPs) are applied on the pre-processed periocular region. Finally, unsupervised discriminant projection (UDP) is used to create the subspaces on WLBP-featured periocular images.

Components of face images across aging can be considered as the age-invariant and age-variant factors. Following this idea, a hidden factor analysis (HFA) model was proposed in [42]. In this method, two latent factors were introduced: an identity factor (age-invariant) and an age factor (age-variant). A linear model was considered to distinct these factors as two subspaces. Also, a learning algorithm was developed using an Expectation Maximization (EM) algorithm to jointly estimate the latent factors and the model parameters. Finally, the cosine distance of the identity components of the gallery and the probe samples are used for face recognition. Note that in [42], the HOG descriptor is used to extract features of face images. Also, due to the existence of noises in the face images, the extracted features are noisy. Hence, the accuracy of identity factor estimation may be reduced. Therefore, in [43], another descriptor, named the maximum entropy feature descriptor (MEFD), has been proposed to improve the representation of face images across aging. This descriptor encodes the microstructure of facial images into a set of discrete codes in terms of maximum entropy in order to extract the appropriate features of the facial images. Then, the method named identity factor analysis (IFA) is used to determine the probability that two face images are related to the same person.

The assumption of independence between the identity and age factors is incorrect, because changes in the face appearance due to aging may vary in different individuals. Hence, in [44], by modifying the HFA method proposed in [42], a probabilistic discriminant method has been proposed to make a better estimate of the identity factor. In this approach, a latent factor is modeled considering the correlation between age and identity factors. Indeed, the person-specific aging information and some features, such as pose and expressions which affect the face recognition task, are jointly modeled by the latent factor. Finally, face recognition is performed using a maximum likelihood approach.

2.3 Deep Neural Network Approaches

As the age variation is a nonlinear and smooth transformation, in [45], a neural network model, named coupled auto-encoder network (CAN), was proposed to overcome the aging in face images. In the proposed CAN, a couple of two auto-encoders have been bridged using two shallow neural networks. Also, in this paper, facial images are decomposed into three components: identity feature, age feature

and noise, using a proposed nonlinear factor analysis method. Among the obtained features from the facial image, the identity feature has been considered as an age-invariant feature to use in the face recognition task.

In general, deep networks have a better performance than shallow networks. The proposed approach in [46] has shown the application of convolutional neural networks (CNNs) in age-invariant face recognition. In this paper, a latent factor guided CNN (LF-CNN) framework was proposed to learn the age-invariant deep face features. The age-invariant deep features are extracted from convolutional features using a designed fully connected layer, named latent factor fully connected (LF-FC) layer. A latent variable model, named latent identity analysis (LIA), was developed to divide the convolutional features into aging (age-variant) and identity (age-invariant) components. The parameters of LF-FC layer are updated using the parameters of the LIA model.

The facial aging process can be well understood considering demographic estimation of facial images, such as estimation of age group, gender and race. Hence, a demographic-assisted face recognition approach was proposed in [47]. This method consists of two main steps: facial-asymmetry-based demographic estimation step and demographic-assisted face recognition step. In the first step, three CNNs are trained for each age group, gender and race classification task in order to extract the demographic features from the query face image. In the second step, at first, deep CNN (dCNN) features are extracted from the query image using VGGNet [48]. Then, dCNN features are used to obtain top k matches from gallery against the query face image. Finally, the top k matched face images are re-ranked using the estimated demographic features.

An age estimation guided convolutional neural network (AE-CNN) has been proposed in [49]. In this approach, an age estimation process is used to separate age-invariant features from those affected by aging. Hence, for extracting the age-invariant features, the AE-CNN contains three fully connected layers. General (age-invariant and age-variant) features are the output of the first fully connected layer. In the second fully connected layer, the softmax loss function is used for estimating the age. The output of this layer is the age features. Also, the third fully-connected layer is used to extract age-specific factors. Finally, age factor is subtracted from the general features to obtain age-invariant features. After that, the softmax loss function is used to recognize the face considering the obtained age-invariant features.

Note that in some face recognition methods, the extracted features may be projected in another space to obtain the comparable features. Hence, in [50], a similarity measure and a distance metric optimization-driven learning approach have been proposed in order to preserve the interaction between the projected features and the used similarity measure. In this method, feature leaning and distance metric learning have been performed at the same time, using a deep convolutional neural network (CNN). To train the CNN, a large number of positive (matched) and negative (unmatched) pairs are generated given the labeled training images. Also, to learn the features which are more independent on aging, the matched pairs are selected considering different ages for each person. Then, using the positive and negative pairs, the CNN is trained subject to reduction and increase of the difference, respectively, between the matched and unmatched pairs. Also, during training the CNN, the parameters of the model are tuned using the mini-batch stochastic gradient descent (SGD) algorithm.

3. AGE INVARIANT FACE RECOGNITION DATABASES

In this section, the databases which are often used in age-invariant face recognition research are briefly introduced. These databases are FGNET [51], MORPH [52], CACD [53], CACD-VS [54] and AgeDB [55]. The FGNET and MORPH databases are commonly used in age-invariant face recognition approaches. Recently, the CACD, CACD-VS and AgeDB databases were introduced for face recognition across aging.

FGNET: The FGNET database includes 1,002 face images from 82 different people. The age range in this database is from 0 to 69 years. The average number of existing face images for each person is approximately 12. The age gap in this database is 0-45 years. Some samples of the face images from this database are shown in Figure 1. In this figure, the age of the persons for whom the face image has been captured is mentioned below the image.

91

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

Figure 1. Sample face images from FGNET. The number below each
facial image represents the age in years.

MORPH: The MORPH database includes 55,134 face images from 13,618 different people. The age range of this database is from 16 to 77 years. The average number of existing face images for each person is approximately 4. The age gap in this database is 0-5 years. Some samples of the face images from this database are shown in Figure 2. In this figure, the age of the person for whom the face image has been captured is mentioned below the image.

Figure 2. Sample face images from MORPH. The number below each
facial image represents the age in years.

The MORPH database has been extended in two albums. Album 1 consists of 1,690 face images from 632 different people. The age range of this album is from 15 to 68 years. The average number of existing face images for each person is approximately 3. Album 2 consists of 78,207 face images from 20,569 different people. The age range of this album is from 15 to 77 years. The average number of existing face images for each person is approximately 4.

CACD: The CACD database includes 163,446 face images from 2,000 different people. The age range of this database is from 16 to 62 years. The average number of existing face images for each person is approximately 82. The age gap in this database is 0-10 years. Some samples of the face images from this database are shown in Figure 3. In this figure, the numbers in the top row are the birth years of the people and the numbers in the left column are the years in which the images have been captured.
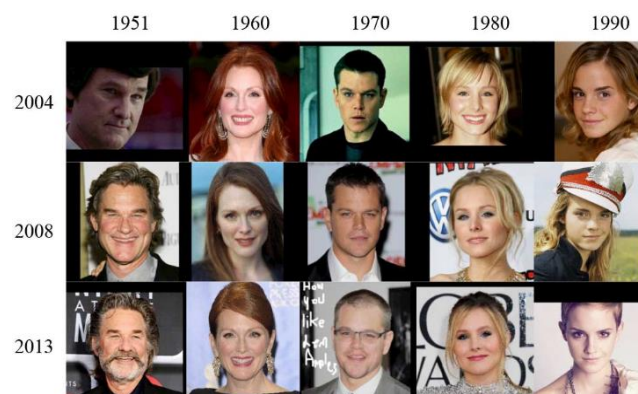
Figure 3. Sample face images from CACD.

CACD-VS: The CACD-VS database is a subset of CACD database which is used for face verification. It contains 4,000 images from 2,000 persons across ages. Indeed, in this database, two images (a positive image and a negative image) exist for each person. In other words, CACD-VS dataset includes 2,000 positive pairs (images from same persons) and 2,000 negative pairs (not from the same persons).

AgeDB: This database contains 16,488 images from 568 different people captured in real-world conditions. Hence, the images of this database may include different poses and expression, noise, occlusions and any uncontrolled conditions. Hence, this database would be used for training and testing the deep neural networks. Also, the age range of this database is from 1 to 101 years. The average number of existing face images and the average age range for each person are 29 and 50.3 years, respectively. Figure 4 shows some sample images from this database.

Figure 4. Sample face images from AgeDB.

4. PERFORMANCE COMPARISON

In this section, the experimental results from some existing age-invariant face recognition methods are reviewed on the FGNET and MORPH databases. The performance of these methods is compared considering the reported Rank-1 recognition accuracy. In general, Rank-k (k = 1, 2, 3, …) is one of the measures used for depicting the performance of face recognition methods. Usually, the recognition rate of the compared methods is considered as a function of Rank-k. Indeed, for ranking the obtained results from a face recognition method, k determines that k-top matches the correct answer [56]. Therefore, for k = 1, Rank-k is the strictest measure; whereas, for k > 1, this measure permits some error. In Table 1, the performance of these methods is compared considering the reported Rank-1 recognition accuracy.

As mentioned before, generative aging models need parameter assumption, landmark point location for each face image and the captured face images in the controlled conditions. Hence, these methods do not work well in real-world face recognition. In practice, the performance of the generative model proposed in [29] is lower than for the other methods mentioned in Table 1.

The method proposed in [37] considers a holistic representation of the facial images. However, aging affects the different components of the face variously. Hence, component-based facial image representation has been proposed in [39], [41]. As it is shown in Table 1, the component-based facial image representation proposed in [41] has achieved 100% Rank-1 recognition accuracy on FGNET database.

Also, separating aging variations from the person-specific features is the main idea in some non-generative methods [42]-[44] to obtain robust age-invariant face features. In these methods, the basic features which are used for estimating the age-invariant and age-variant factors play an important role in the final results. As mentioned before, in the method proposed in [43], a MEFD descriptor has been introduced to be used instead of HOG. Hence, according to Table 1, this method achieves better accuracy in comparison with the similar methods proposed in [42], [44].

As mentioned before, the methods proposed in [45]-[47], [49]-[50] are the recently developed approaches in age-invariant face recognition. In these methods, using the deep learning approaches leads to obtain the deep features in the facial images which are aging-invariant.

Age variation is a nonlinear and smooth transformation. However, in [42]-[44], a linear factor analysis method is used in order to separate the age-variant and age-invariant factors from the face images. Hence, this method has lower performance when compared to the methods proposed in [45]-[46] and [49]-[50] (see Table 1). Besides, the kind of basic descriptor used for feature extraction in [42]-[44] can be the other reason why the results of these methods are not better than those of deep learning-based approaches.

Also, the methods proposed in [46], [49] have better performance in comparison with the method proposed in [45]. Indeed, the unsupervised CAN model proposed in [45] is less discriminative than the supervised LF-CNNs and AE-CNN proposed in [46], [49]. Also, the deep networks used in [46], [49] have better performance than the shallow networks used in [45].

Apparently different components of the face are affected variously across aging progression. Hence, age-invariant face recognition approaches which consider this issue outperform the other approaches. Besides, separating the identity and aging factors of the facial images increases the accuracy of the face recognition approaches across aging. Indeed, we can conclude from Table 1 that face recognition approaches will recognize the face more accurately compared to the other methods, if they consider component-based representation of facial images and consequently estimate the identity factors from aging factors in each component. Also, deep learning techniques have emerged in machine vision applications [57] and recently, age-invariant face recognition methods tend to use deep learning approaches. Hence, the capability of these neural networks in feature extraction and modeling can be employed to estimate the identity and aging factors in each component of facial images. Therefore, a combination of the mentioned approaches may lead to an appropriate performance of age-invariant face recognition approach.

Table 1. Experimental results of the existing age-invariant face recognition methods.

The method proposed in	Databases (# subjects, # images) in probe and gallery	Reported Rank-1 recognition accuracy
[29]	FGNET (82, 1002)	37.4%
	MORPH album 1 (612, 612)	66.4%
	MORPH album 2 (10000, 20000)	79.8%
[37]	FGNET (82,1002)	47.50%
	MORPH album 2 (10000, 20000)	83.90%
[39]	MORPH album 2 (10000, 20000)	81.27%
[41]	FGNET (82, 1002)	100%
[42]	FGNET (82, 1002)	69.0%
	MORPH album 2 (10000, 20000)	91.14%
[43]	FGNET (82, 1002)	76.2%
	MORPH album 2 (10000, 20000)	93.80%
[44]	FGNET (82, 1002)	72.8%
	MORPH album 2 (10000, 20000)	87.94%
[45]	FGNET (82, 1002)	86.5%
[46]	FGNET (82, 1002)	88.1%
	MORPH album 2 (10000, 20000)	97.51%
[49]	MORPH album 2 (10000, 20000)	98.13%
[50]	MORPH album 2 (10000, 20000)	93.6%

5. CONCLUSIONS

Face recognition has applications in many machine vision fields, such as law enforcement and forensic investigation, homeland security, missing persons and checking the passport and visa. In these applications, the images in database may not be updated continuously. As aging slowly leads to changing the important features of the facial image, it directly affects the performance of face recognition approaches. Hence, it is necessary to consider the aging variations in face recognition. There are some research studies which have considered the face aging progression. These approaches have tried to estimate and simulate aging and recognize the face. In this paper, some age-invariant face recognition approaches have been briefly reviewed. Also, the experimental results of these methods have been compared on a number of databases. The experimental results show that age-invariant face recognition approaches which consider both component-based representation of facial images and identity factors have better performance compared with the other face recognition methods across aging.

"A Survey on Age-Invariant Face Recognition Methods", Z. Mortezaie and H. Hassanpour

REFERENCES

[1]   M. Chihaoui, A. Elkefi, W. Bellil and C. Ben Amar, "A Survey of 2D Face Recognition Techniques," Computers, vol. 5, no. 4, p. 21, 2016.

[2]   M. Sharif, F. Naz, M. Yasmin, M. A. Shahid and A. Rehman," Face Recognition: A Survey," Journal of Engineering Science & Technology Review, vol. 10, no. 2, 2017.

[3]   M. M. Sawant and K. M. Bhurchandi, "Age Invariant Face Recognition: A Survey on Facial Aging Databases, Techniques and Effect of Aging," Artificial Intelligence Review, pp. 1-28, 2018.

[4]   A. K. Agrawal and Y. N. Singh, "Evaluation of Face Recognition Methods in Unconstrained Environments," Procedia Computer Science, vol. 48, pp. 644-651, 2015.

[5]   S. Sahni and S. Saxena, "Survey: Techniques for Aging Problems in Face Recognition," MIT International Journal of Computer Science and Information Technology, vol. 4, no. 2, pp. 82-88, 2014.

[6]   L. Best-Rowden and A. K. Jain, "Longitudinal Study of Automatic Face Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 1, pp. 148-162, 2018.

[7]   H. Liao, Y. Yan, W. Dai and P. Fan, "Age Estimation of Face Images Based on CNN and Divide-and-Rule Strategy," Mathematical Problems in Engineering, 2018.

[8]   H. Liu, J. Lu, J. Feng and J. Zhou, "Group-aware Deep Feature Learning for Facial Age Estimation," Pattern Recognition, vol. 66, pp. 82-94, 2017.

[9]   K. Y. Chang and C. S. Chen, "A Learning Framework for Age Rank Estimation Based on Face Images with Scattering Transform," IEEE Transactions on Image Processing, vol. 24, no. 3, pp. 785-798, 2015.

[10]   J. Lu, V. E. Liong and J. Zhou,"Cost-sensitive Local Binary Feature Learning for Facial Age Estimation," IEEE Transactions on Image Processing, vol. 24, no. 12, pp. 5356-5368, 2015.

[11]   X. Yang, B. B. Gao, C. Xing, Z. W. Huo, X. S. Wei, Y. Zhou, J. Wu and X. Geng, "Deep Label Distribution Learning for Apparent Age Estimation," Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 102-108, 2015.

[12]   I. Huerta, C. Fernández, C. Segura, J. Hernando and A. Prati, "A Deep Analysis on Age Estimation," Pattern Recognition Letters, vol. 68, pp. 239-249, 2015.

[13]   J. Ylioinas, A. Hadid, X. Hong and M. Pietikäinen, "Age Estimation Using Local Binary Pattern Kernel Density Estimate," Int. Conf. on Image Analysis and Processing, pp. 141-150, 2013.

[14]   A. Montillo and H. Ling, "Age Regression from Faces Using Random Forests," International Conference on Image Processing (ICIP), pp. 2465-2468, 2009.

[15]   G. Guo, G. Mu, Y. Fu and T. S. Huang, "Human Age Estimation Using Bio-inspired Features," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 112-119, 2009.

[16]   Y. Fu and T. S. Huang, "Human Age Eestimation with Regression on Discriminative Aging Manifold," IEEE Transactions on Multimedia, vol. 10, no. 4, pp. 578-584, 2008.

[17]   S. K. Zhou, B. Georgescu, X. S. Zhou and D. Comaniciu, "Image-based Regression Using Boosting Method," Proc. of the IEEE International Conference on Computer Vision (ICCV), vol. 1, pp. 541-548, 2005.

[18]   G. Guo, Y. Fu, C. R. Dyer and T. S. Huang, "Image-based Human Age Estimation by Manifold Learning and Locally Adjusted Robust Regression," IEEE Transactions on Image Processing, vol. 17, no. 7, pp. 1178-1188, 2008.

[19]   X. Geng, Z. H. Zhou and K. Smith-Miles, "Automatic Age Estimation Based on Facial Aging Patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 12, pp. 2234-2240, 2007.

[20]   Y. H. Kwon and N. da Vitoria Lobo, "Age Classification from Facial Images," Computer Vision and Image Understanding, vol. 74, no. 1, pp. 1-21, 1999.

[21]   J. Suo, X. Chen, S. Shan and W. Gao, "Learning Long Term Face Aging Patterns from Partially Dense Aging Databases," Proc. of International Conference on Computer Vision, pp. 622-629, 2009.

[22]   A. Lanitis, C. J. Taylor and T. F. Cootes, "Toward Automatic Simulation of Aging Effects on Face Images," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 24, no. 4,pp. 442-455, 2002.

[23]   J. Suo, S. C. Zhu, S. Shan and X. Chen, "A Compositional and Dynamic Model for Face Aging," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 3, pp. 385-401, 2010.

[24]   N. Tsumura, N. Ojima, K. Sato, M. Shiraishi, H. Shimizu, H. Nabeshima, S. Akazaki, K. Hori and Y. Miyake, "Image-based Skin Color and Texture Analysis/Synthesis by Extracting Hemoglobin and Melanin Information in the Skin," ACM Trans. on Graphics (TOG), vol. 22, no.3, pp. 770-779, 2003.

[25]   J. X. Du, C. M. Zhai and Y. Q. Ye, "Face Aging Simulation Based on NMF Algorithm with Sparseness Constraints," International Conference on Intelligent Computing, pp. 516-522. Springer, Berlin, Heidelberg, 2011.

[26]   J. Wang, Y. Shang, G. Su and X. Lin, "Age Simulation for Face Recognition," Proc. of International Conference on Pattern Recognition (ICPR), vol. 3, pp. 913-916, 2006.

[27]   N. Ramanathan and R. Chellappa, "Modeling Shape and Textural Variations in Aging Faces," Proc. of IEEE International Conference on Automatic Face & Gesture Recognition, pp. 1-8, 2008.

[28]   J. X. Du, C. M. Zhai and Y. Q. Ye, "Face Aging Simulation and Recognition Based on NMF Algorithm with Sparseness Constraints," Neurocomputing, vol. 116, pp. 250-259, 2013.

[29]   U. Park, Y. Tong and A. K. Jain, "Age-invariant Face Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 5, pp. 947-954, 2010.

[30]   P. O. Hoyer, "Non-negative Matrix Factorization with Sparseness Constraints," Journal of Machine Learning Research, pp. 1457-1469, 2004.

[31]   G. Mahalingam and C. Kambhamettu, "Age Invariant Face Recognition Using Graph Matching," Proc. of the IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS), pp. 1-7, 2010.

[32]   P. S. Penev and J. J Atick, "Local Feature Analysis: A General Statistical Theory for Object Representation," Networks: Computation in Neural Systems, vol. 7, no. 3, pp. 477-500, 1996.

[33]   R. Duda, P. Hart and D. Stork, Pattern Classification, John Wiley & Sons, New York, 2nd Edition, 2001.

[34]   N. Ramanathan and R. Chellappa, "Face Verification across Age Progression," IEEE Transactions on Image Processing, vol. 15, no. 11, pp. 3349-3361, 2006.

[35]   H. Ling, S. Soatto, N. Ramanathan and D. W. Jacobs, "Face Verification across Age Progression Using Discriminative Methods," IEEE Transactions on Information Forensics and Security, vol. 5, no. 1, pp. 82-91, 2010.

[36]   H. Ling, S. Soatto, N. Ramanathan and D. W. Jacobs, "A Study of Face Recognition As People Age," Proc. of the International Conference on Computer Vision (ICCV), pp. 1-8, 2007.

[37]   Z. Li, U. Park and A. K. Jain, "A Discriminative Model for Age Invariant Face Recognition," IEEE Transactions on Information Forensics and Security, vol. 6, no. 3, pp. 1028-1037, 2011.

[38]   D. Sungatullina, J. Lu, G. Wang and P. Moulin, "Multiview Discriminative Learning for Age-invariant Face Recognition," Proc. of the International Conference and Workshops on Automatic Face and Gesture Recognition (FG), pp. 1-6, 2013.

[39]   C. Otto, H. Han and A. Jain, "How Does Aging Affect Facial Components?," European Conference on Computer Vision, pp. 189-198, Springer, Berlin, Heidelberg, 2012.

[40]   T. F. Cootes, C. J. Taylor, D. H. Cooper and J. Graham, "Active Shape Models-Their Training and Application," Computer Vision and Image Understanding, vol. 61, no. 1, pp. 38-59, 1995.

[41]   F. Juefei-Xu, K. Luu, M. Savvides, T. D. Bui and C. Y. Suen, "Investigating Age Invariant Face Recognition Based on Periocular Biometrics," Proc. of the Int. Joint Conf. on Biometrics, pp. 1-7, 2011.

[42]   D. Gong, Z. Li, D. Lin, J. Liu and X. Tang, "Hidden Factor Analysis for Age Invariant Face Recognition," Proc. of IEEE Int. Conf. on Computer Vision (ICCV), pp. 2872-2879, 2013.

[43]   D. Gong, Z. Li, D. Tao, J. Liu and X. Li, "A Maximum Entropy Feature Descriptor for Age Invariant Face Recognition," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 5289-5297, 2015.

[44]   H. Li, H. Zou and H. Hu, "Modified Hidden Factor Analysis for Cross-age Face Recognition," IEEE Signal Processing Letters, vol. 24, no. 4, pp. 465-469, 2017.

[45]   C. Xu, Q. Liu and M. Ye, "Age Invariant Face Recognition and Retrieval by Coupled Auto-encoder Networks," Neurocomputing, vol. 222, pp. 62-71, 2017.

[46]   Y. Wen, Z. Li and Y. Qiao, "Latent Factor Guided Convolutional Neural Networks for Age-invariant Face Recognition," Proc. of IEEE Conf. on Comp. Vision and Pattern Recog., pp. 4893-4901, 2016.

[47]   M. Sajid, T. Shafique, S. Manzoor, F. Iqbal, H. Talal, U. Samad Qureshi and I. Riaz, "Demographic-Assisted Age-invariant Face Recognition and Retrieval," Symmetry, vol. 10, no. 5, pp. 1-17, 2018.

[48]   O. M. Parkhi, A. Vedaldi and A. Zisserman, "Deep Face Recognition," Proceedings of the British Machine Vision Conference (BMVC), vol. 1, no. 3, p. 6, Swansea, UK, 2015.

[49]   T. Zheng, W. Deng and J. Hu, "Age Estimation Guided Convolutional Neural Network for Age-invariant Face Recognition," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1-9, 2017.

[50]   Y. Li, G. Wang, L. Nie, Q. Wang and W. Tan, "Distance Metric Optimization Driven Convolutional Nneural Network for Age Invariant Face Recognition," Pattern Recognition, vol. 75, pp. 51-62, 2018.

[51]   T. Cootes, "FGNET Face and Gesture Recognition Database," Face and Gesture Recognition Working Group, [Online], Available at: http://www-prima.inrialpes.fr/FGnet/

[52]   K. Ricanek and T. Tesafaye, "Morph: A Longitudinal Image Database of Normal Adult Age-Progression," Proc. of the Int. Conf. on Aut. Face and Gesture Recognition (FGR), pp. 341-345, 2006.

[53]   B. C. Chen, C. S. Chen and W. H. Hsu, "Cross-age Reference Coding for Age-invariant Face Recognition and Retrieval," Proc. of the European Conference on Computer Vision, pp. 768-783, Springer, Cham, 2014.

[54]   B. C. Chen, C. S. Chen and W. H. Hsu, "Face Recognition and Retrieval Using Cross-age Reference Coding with Cross-age Celebrity Dataset," IEEE Transactions on Multimedia, vol. 17, no. 6, pp. 804-815, 2015.

[55]   S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia and S. Zafeiriou, "AgeDB: The First Manually Collected, in-the-Wild Age Database," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 51-59, 2017.

[56]   K. Delac, M. Grgic and S. Grgic,"Statistics in Face Recognition: Analyzing Probability Distributions of PCA, ICA and LDA Performance Results," Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA), pp. 289-294, 2005.

[57]   M. Wang and W. Deng, "Deep Face Recognition: A Survey," arXiv preprint arXiv:1804.06655, 2018.

ملخص البحث:

يســتخدم تمييــز الوجــوه فــي كثيــر مــن التطبيقــات، لاســيما الأمنيــة والرقابيــة منهــا. وهنالــك بعــض المســائل، مثــل: التقــدم فــي العمــر، والإطبــاق الجزئــي للأســنان، والتغيــر فــي وضــع التصــوير والإضــاءة، وتعبيــر الوجــه، التــي تــؤثر علــى نحـو مباشــر فــي الطــرق المســتخدمة فــي تمييــز وجــوه الأشــخاص. وفــي العــادة، بالعديــد مــن التطبيقــات، مثــل فحــص جــوازات وتأشــيرات الســفر، لا يــتم تحــديث الصــور فــي قواعــد البيانــات بإســتمرار. فــي مثــل هــذه الحــالات، يــؤدي التقــدم فــي العمــر إلــى تغييــر الســِّمات المهمــة لصــورة الوجــه. لــذا فــإن تمييــز الوجــوه مــن المســائل الجوهريــة، وخصوصــاً فــي أنظمــة الأمن والمراقبة.

تقــدم هــذه الورقــة وصفــاً مــوجزاً لعــددٍ مــن طــرق تمييــز الوجــوه مــن حيــث متانتهــا بالنســبة للتغيــرات التــي تطــرأ علــى وجــوه الأشــخاص بفعــل التقــدم فــي العمــر. مــن ناحيــة أخــرى، تمــت مقارنــة نتــائج هــذه الطــرق عبــر تطبيقهــا علــى عــدد مــن قواعــد البيانــات شــائعة الاســتخدام فــي عمليــة تمييــز وجــوه الأشــخاص. وقــد بينــت النتــائج أنّ الطــرق التــي تأخــذ بعــين الاعتبــار تمثيــل صــورة الوجــه المبنــي علــى تقســيمها إلــى مكونــات قــد تفوقــت علــى غيرها من الطرق المستخدمة من حيث دقة الأداء في تمييز وجوه الأشخاص.

97

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

CONVOLUTIONAL NEURAL NETWORK MULTI-EMOTION CLASSIFIERS

S. S. Ibrahiem[1], S. S. Ismail [2], K. A. Bahnasy[3] and M. M. Aref [4]

ABSTRACT

Natural languages are universal and flexible, but cannot exist without ambiguity. Having more than one attitude and meaning in the same phrase context is the main cause for word or phrase ambiguity. Most previous work on emotion analysis has only covered single-label classification and neglected the presence of multiple emotion labels in one instance. This paper presents multi-emotion classification in Twitter based on Convolutional Neural Networks (CNNs). The applied features are emotion lexicons, word embeddings and frequency distribution. The proposed networks performance is evaluated using state-of-the-art classification algorithms, achieving a hamming score range from 0.46 to 0.52 on the challenging SemEval2018 Task E-c.

KEYWORDS

Emotion classification, Multi-label classification, Convolutional neural network, Twitter.

1. INTRODUCTION

Online social media, such as Twitter, can communicate people's facts, opinions and emotions on different kinds of topics in short texts. Analyzing the emotions expressed in these texts has attracted researchers in the natural language processing research field. Emotion analysis is the task of determining the feeling or attitude towards a target or topic. It has a wide range of applications in politics, public health, commerce and business. Many real-world problems can be modeled by multi-label classification systems, like emotion analysis, since one tweet instance can imply more than one emotion. Traditional emotion analysis methods are single-label classification-based, while multi-label classification (MLC) recently attracts researchers' interest due to its applicability to a wide range of domains [1]-[3]. One of the most common used approaches is problem transformation methods that transform a multi-label dataset into a single-label dataset, so that existing single-label classifiers can be applied to multi-label datasets. Problem transformation approach replaces each multi-label instance with a single-class label for each class-label occurring. Binary relevance is the most common used method in the problem transformation approach; it works by decomposing the multi-label learning task into a number of independent binary learning tasks [2]-[4]. It suffers from directly modeling correlations that may exist among labels.

This paper is going to propose multi-emotion classification in Twitter based on Convolutional Neural Networks (CNNs). It performs two experiments to compare the proposed architecture with the state-of-art emotion classification approaches. The experimental results achieved a high classification accuracy and outperformed the state-of-the art approaches. The paper is organized in sections as follows. Section 2 discusses the research background and related work. Then, Section 3 discusses the applied datasets and lexicons. Section 4 discusses the proposed emotion detection approach and its evaluation. Section 5 concludes the paper.

2. RELATED WORK

Traditional approaches to sentiment analysis and emotion detection were categorized into lexicon-based methods (Y. Cao et al. [5], L. Flekova et al. [6], D. M. El-Din et al. [7], S. Mohamed et al. [8]), supervised machine learning methods (E. Cambria et al. [9], Y. Wang et al. [10], P. Sobhani et al. [11], N. Majumder et al. [12], R. Oramas et al. [13], M. Suhasini et al. [14]) and hybrid methods (S.

---

1. S. S. Ibrahiem is PhD researcher with Department of Computer Science, Ain Shams University, Cairo, Egypt. Email: soha.elshafey@cis.asu.edu.eg
2. S. S. Ismail is Computer Science Teacher, Ain Shams University, Cairo, Egypt. Email: sallysaad@cis.asu.edu.eg
3. K. A. Bahnasy is Information System Professor, Ain Shams University, Cairo, Egypt. Email: khaled.bahnasy@oi.edu.eg
4. M. M. Aref is Computer Science Professor, Ain Shams University, Cairo, Egypt. Email: mostafa.aref@cis.asu.edu.eg

Mohamed et al. [15], X. Sun et al. [16], B. Gaind et al. [17]).

Y. Cao et al. [5] focused on the task of disambiguating polarity-ambiguous words and the task was reduced to sentiment classification of aspects, which referred to sentiment expectation instead of semantic orientation. In order to disambiguate polarity-ambiguous words, the research constructed the aspect-and polarity-ambiguous lexicon using a mutual bootstrapping algorithm. So, the sentiment of polarity-ambiguous words in context could be decided collaboratively by the sentiment expectation of the aspect-and polarity-ambiguous words' prior polarity. Training corpus was 6000 positive and negative reviews about computer and books, while testing corpus was 1000 reviews. The average F1-measures were 75% in books and 79% in computer reviews.

L. Flekova et al. [6] introduced a method to identify frequent bigrams where a word switches polarity and to find out which words were bipolar to the extent, so that it was better to have them removed from the polarity lexica. The introduced method demonstrated that the scores match human perception of polarity and bring improvement in the classification results using its enhanced context-aware method. It enhanced the assessment of lexicon-based sentiment detection algorithms and could be used to quantify ambiguous words. 1600 Facebook messages were annotated by positive and negative sentiments that were used to asset the lexicon's performance on different feature sets. The accuracy ranged from 66% to 76%.

D. M. El-Din et al. [7] proposed a new technique to analyze online reviews. It was called sentiment analysis of online papers (SAOOP). SAOOP was a new technique used for enhancing bag-of-words model, improving accuracy and performance. SAOOP was useful in increasing the understanding rate of review's sentences through higher language coverage cases. SAOOP introduced solutions for some sentiment analysis challenges and used them to achieve higher accuracy. Two datasets were used; real dataset which splits into two datasets with training set (1000 text reviews) and test set (5000 text reviews) and the verified data set (10.000 text reviews) which included more than 5.000 positive words dataset and 5.000 negative words. The accuracy of SAOOP was 82%.

S. Mohamed et al. [8] presented a data-driven study comparing the emotionality of metaphorical expressions with that of their literal counterparts. Its results indicated that metaphorical usages are, on average, significantly more emotional than literal ones. It also showed that this emotional content was not simply transferred from the source domain into the target, but rather is a result of meaning composition and interaction of the two domains in the metaphor. It used 1639 senses of 440 verbs in WordNet. The confidence was 95%.

E. Cambria et al. [9] introduced a vector space model which was built by means of random projection to allow reasoning by natural language concepts. The model allowed semantic features associated with concepts to be generalized and to be intuitively clustered according to their semantic and affective relatedness. Such an affective intuition enabled the inference of emotions and polarity conveyed by multi-word expressions, thus achieving efficient concept-level sentiment analysis. An affective common-sense knowledge is built by applying concept frequency - inverse opinion frequency (CF-IOF) on a 5,000- blogpost database extracted from LiveJournal1, that is category-and mood-labeled by users. Test dataset was 2000 manually tagged patient reviews associating to each a category service. F-measure ranged from 74% to 85.1% according to evaluated service.

Y. Wang et al. [10] proposed a constraint optimization framework to discover emotions from social media content of the users. This framework employed several novel constraints, such as emotion bindings, topic correlations, along with specialized features proposed by prior work and well-established emotion NRC[2] lexicons[3]. It proposed an efficient inference algorithm and reported promising empirical results on three diverse datasets. Another distinguishing feature of this model was that it solved multi-label classification problem and allowed a document to have multiple emotions. The evaluated datasets were SemEval[4] of 1250 news headlines with an average F-measure of 0.63,

---

[1] http://livejournal.com/
[2] National Research Canada (NRC)
[3] http://saifmohammad.com/WebPages/lexicons.html
[4] http://web.eecs.umich.edu/~mihalcea/downloads.htm

ISEAR[5] of 7666 sentences annotated by 1096 participants with different cultural backgrounds with an average F-measure of 0.74 and a Twitter dataset of 1800 tweets using the Twitter API with an average F-measure of 0.522.

P. Sobhani et al. [11] developed a simple stance detection system that outperforms all 19 teams that participated in a recent shared task competition on the same dataset (SemEval-2016 Task #6). It applied n-grams, NRC lexicons, word embeddings and support vector machine learning. The classification range was in favour or against classes. The automatic system evaluation F1-measure was 70.32%.

N. Majumder et al. [12] presented a method to extract personality traits from a stream of consciousness essays using a convolutional neural network (CNN). It has been trained on five different networks, all with the same architecture, for the five studied personality traits. Each network was a binary classifier that predicted the corresponding trait to be positive or negative. It developed a novel document modeling technique based on a CNN feature extractor. Namely, it had been fed sentences from the essays to convolution filters in order to obtain the sentence model in the form of n-gram and word embedding feature vectors. Each individual essay was represented by aggregating the vectors of its sentences. For final classification, the document vector was fed into a fully connected neural network with one hidden layer and the final softmax layer of two sizes, representing the yes and no classes. 50 epochs for training and tenfold cross-validation were used to evaluate the trained network. The network was evaluated to SVM and multi-layer perception learning and the accuracy ranged from 50% to 62%.

R. Oramas et al. [13] created a corpus of phrases (opinions) and categorized them into frustration, boring, excitement and engagement phrases. The corpus was tested using several tests with different classifiers: Multi-nomial Naive Bayes classifier, Support Vector Machine, Linear Support Vector Machine, Stochastic Gradient Descent Classifier and K-Nearest Neighbors classifier. The used dataset consisted of 851 opinions. The classifier with the highest score was Bernoulli Naive Bayes classifier with an accuracy of 76.77%.

M. Suhasini et al. [14] proposed a method which detected the emotion or mood of the tweets and classified the Twitter messages under appropriate emotional categories. The method used was a two-approach method. The approach used were the Rule-Based approach and the Machine Learning approach. First approach contributed in pre-processing, tagging, feature selection and knowledge base creation. Rule-based approach was used to classify the tweets under four class categories (Happy-Active, Happy-Inactive, Unhappy-Active and Unhappy-Inactive). The second approach was based on a supervised machine learning algorithm called Naïve Bayes, which requires labeled data. The rule-based approach was able to classify the tweets with an accuracy around 85% and with the machine learning approach the accuracy was around 88%.

S. Mohamed et al. [15] automatically annotated a set of 2012 US presidential election tweets for a number of attributes pertaining to sentiment, emotion, purpose and style by crowdsourcing. Overall, more than 100,000 crowdsourced responses were obtained for 13 questions on emotions, style and purpose. Additionally, it was shown through an analysis of these annotations that purpose, even though correlated with emotions, was significantly different. Finally, it was described how automatic classifiers had been developed, using features from state-of-the-art sentiment analysis systems, to predict emotion and purpose labels, respectively, in new unseen tweets. These experiments resulted in an accuracy of 56.84% for automatic systems on this new data.

X. Sun et al. [16] presented a method for extracting emotional elements containing emotional objects and emotional words and their tendencies from product reviews based on a mixed model. First, some conditional random fields were constructed to extract emotional elements, lead-in semantics and word meanings as features to improve the robustness of feature template and rules were used for hierarchical filtering of errors. Then, a support vector machine was constructed to classify the emotional tendencies of the fine-grained elements to achieve key information from product reviews. Deep semantic information was imported based on a neural network to improve the traditional bag-of-

---

[5] http://www.affective-sciences.org/researchmaterial

word model. Experimental results showed that the proposed model with deep features efficiently improved the F-measure 50-80%.

B. Gaind et al. [17] proposed two approaches to classify social media texts into six categories of emotion: Happiness, Sadness, Fear, Anger, Surprise and Disgust. First approach extracted emotion in the texts using natural language processing, like emoticons, part of speech, negations and grammatical analysis. Second approach was based on two machine learnings, which are support to vector machine and J48 classifiers. A large bag of words in English was created that expressed word emotions in addition to their intensities. The training accuracy of the support vector machine was 91.7% and the training accuracy of J48 classifier was 85.4% of 900 tweets.

Through the research studies that have been listed, it can be concluded that firstly there is still insufficient inadequacy of models to explore emotions from the texts of social media due to data size, text structure's context or emotional granularity [4]. Secondly, emotion analysis is modeled as a supervised multi-label classification problem, because one instance may contain one or more emotions from a standard emotion set. This paper deals with multi-emotion classification based on Convolutional Neural Networks (CNNs). This network can keep complementary information and will bring higher accuracy with the assistance of different feature configurations that could lead to possible directions of further improvement.

3. DATASETS & LEXICONS

The datasets and lexicons that are going to be used in this research are listed as follows.

SemEval[6]-2018 is a group of datasets that include an array of subtasks, where automatic systems have to infer the affectual state of a person from his/her tweets. One of its tasks is Emotion Classification (E-c), where a given tweet is classified as 'neutral or no emotion' or as one or more of eleven given emotions that best represent the mental state of the user [18].

The NRC Emotion Lexicon is a list of English words and their associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy and disgust) and two sentiments (negative and positive). The annotations were manually done by crowdsourcing [19]-[20].

NRC Hashtag Emotion Lexicon: it is an association of words with eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy and disgust) generated automatically from tweets with emotion-word hashtags such as #happiness and #anger [21].

NRC VAD Lexicon: The NRC Valence, Arousal and Dominance (VAD) Lexicon includes a list of more than 20,000 English words and their valence, arousal and dominance scores. Valence is the positive and negative or pleasure and displeasure dimension [22]. Arousal is the excited and calm or active and passive dimension [22]. Dominance is the powerful and weak or 'have full control' and 'have no control' dimension [22].

4. EMOTION DETECTION APPROACH

The supervised machine learning approach involves two processes; training process where an algorithm learns from a labeled data and testing process or evaluation process where the algorithm makes predictions on a sample data. In this research, the input tweets are firstly pre-processed and normalized, where different types of processes are applied to denoise and filter important tokens in the tweets (see Figure 1). Secondly, the pre-processed tokens are transformed to feature vectors. These featured vectors exploit contextual and semantic relations between tokens and each target emotion space using the prementioned lexicons and datasets, in addition to word frequency distribution and word embeddings. The word frequency distribution calculates each token's occurrence in the given dataset SemEval-2018 towards a given emotion space [23]. The applied word embedding method is word2vec that works on grouping each emotion's representative tokens together in the same vector space [24]-[25]. Figure 2 illustrates the procedure of transforming the tweets' pre-processed tokens

---

[6] Semantic Evaluation Workshop, 2018.

into feature vectors in both training and testing phases. Finally, the produced tweets' feature vectors are trained and evaluated on the following proposed network.

- Emojis are translated, and URLs are removed from the input tweet.
- The tweets are tokenized.
- Negation is handled, so that context meaning and attitude are conserved.
- Long words are corrected without context loss.
- Any spaces and punctuation letters are filtered, so that it doesn't adversely affect the efficiency of the classification.
- Part of speech tagging is applied to tokens.
- Spelling correction is applied to each token. Any meaningless tokens are neglected.
- Tokens are stemmed from wordnet lexicon, according to their part of speech tag.
- Stop words are removed from tokens. It doesn't imply any emotion.

Figure 1. Pre-processing steps sequence.

```
Input:
        train_tweets_tokens=load_train_tweets_tokens()
        test_tweets_tokens=load_test_tweets_tokens()
Output:
        //Initialize empty output lists that represent the relevance of each tweet to a target emotion class
        anger_frequency,fear_frequency,joy_frequency,sad_frequency=[],[],[],[]
        anger_w2v,fear_w2v,joy_w2v,sad_w2v=[],[],[],[]
        anger_wrdlex,fear_wrdlex,joy_wrdlex,sad_wrdlex=[],[],[],[]
        anger_hshlex,fear_hshlex,joy_hshlex,sad_hshlex=[],[],[],[]
        valence,arousal,dominance=[],[],[]
------------------------------------------------------------------------------------------
//frequency distribution calculation and vector space generation to each emotion class from training tokens only
for each emotion tokens in train_tweets_tokens do
        Compute Frequency distribution of tokens occurred in each emotion class
        output lists of most frequent training tokens in each emotion class
        Compute each emotion vector space from tokens occurred in each emotion tweets
        output each emotions' representative training tokens together in a same vector space
end for;
for each tweet in train_tweets_tokens do
        for each token in tweet:
                //frequency distribution
                Get Frequency distribution of the input token to each training emotion frequent tokens
                Sum up frequency distribution of tokens in each emotion class in different variables
                //word embeddings
                Get token similarity to each emotion training vector space
                Sum up tokens similarities in each emotion class in separate variables
                //NRC word and hash lexicons
                Search word lexicon if the token belongs to an emotion class, increment the word emotion counter of the tweet
                Search hash lexicon if the token belongs to an emotion class, increment the hash emotion counter of the tweet
                //NRC VAD Lexicon
                Search VAD lexicon for the token, get three dimensions of the token valence, arousal, and dominance
                Sum up tokens three dimensions
        end for;

        Calculate average of frequency distributions of the tweet tokens in each emotion class
        Save value of frequency distribution of tokens in each tweet towards each emotion class lists
        in anger_frequency, fear_frequency, joy_frequency, and sad_frequency

        Calculate average similarities of the tweet tokens in each emotion class
        Save similarities values of each tweet towards each emotion class in
        anger_w2v, fear_w2v, joy_w2v, and sad_w2v lists

        Calculate average word lexicon of each emotion counter
        Calculate average hash lexicon of each emotion counter
        Save each average counters each tweet towards each emotion class in
        anger_wrdlex, fear_wrdlex, joy_wrdlex ,sad_wrdlex  lists and
        anger_hshlex,fear_hshlex,joy_hshlex,sad_hshlex lists

        Calculate average dimensions of tweets tokens
        Save average dimensions of each tweet in separate lists valence,arousal, and dominance
end for;
```

Figure 2. Feature generation procedure steps.

4.1 Building and Training Models

Convolutional Neural Networks (CNNs) are one of the most successful network architectures in state-of-the-art Artificial Neural Network (ANN) algorithms. A CNN can learn relevant features from the input text at different levels like the human brain. Its basic components are listed as follows. The convolution is over an input pixel matrix. A kernel or filter slides over the input matrix creating an entry in the activation map for each window in the input matrix. Hereby, the weights of the filter are multiplied by the values in the window of the input matrix and the results are added up. The weights in the filter are subject to the learning process of the network and are shared over all windows of the convolution operation. In CNN architectures, a convolution layer is usually followed by a pooling layer. Pooling layers sub-sample their input and can be applied over the whole matrix or over windows. They significantly reduce the output dimensionality without losing much information [26].

In this paper, a Convolutional Neural Network (CNN) is built and trained to predict multi-emotion labeled tweets. The output emotion classes are anger, fear, joy and sadness. The SemEval2018 task e-c dataset is divided into training and testing datasets. The training dataset consists of 5000 labeled tweets, while the testing dataset consists of 1000 annotated tweets. Two experiments are applied on two feature configurations. Figures 3 and 5 show two different CNN network architectures according to the applied feature configuration. In the first experiment, the network has five input layers. The first four input layers are of size four, representing the tweet's features for a target emotion using frequency distribution, word-to-vector similarity, NRC word emotion lexicon and NRC hashtag lexicon. The fifth input layer is of size three, which represents the tweet's average valence-arousal-dominance feature using NRC VAD lexicon. The first four input convolutional layers have 4 filters and window size 2, followed by an average pooling layer with pool size 3. The fifth input layer has three filters and window size 2, followed by an average pooling size 2. Flatten layers work as a connection between convolutional layer output and the following dense layers. This classifier has 2 fully connected dense layers with sizes of 10 and 4 nodes, respectively. The output node represents the probability of likelihood of the input tweet vector to the four emotions: anger, fear, joy and sadness.
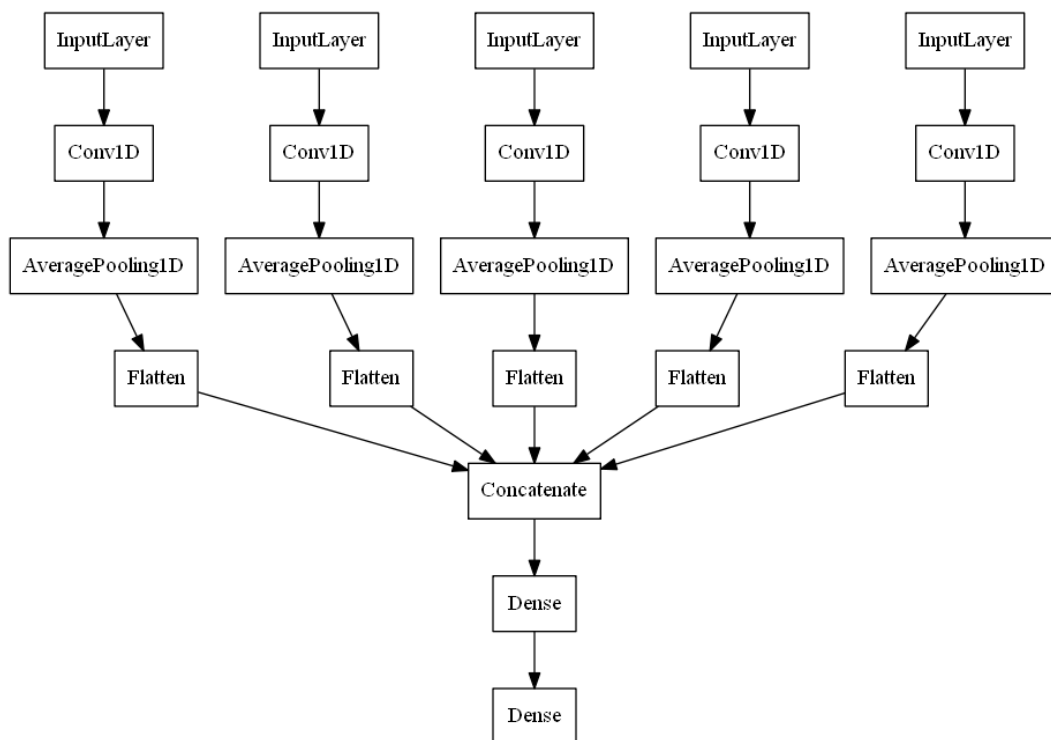
Figure 3. Large-feature CNN emotion classification architecture.

The training dataset is transformed to the defined feature vectors. These vectors are trained on the presented network to produce the emotion prediction model. The tools used are Keras [27], Python [28]-[29] and NLTK [30]. Figure 4 shows the CNN model accuracy and loss graphs during the training phase using a 0.34 cross-validation set. Cross-validation technique is used to evaluate

predictive models by partitioning the original sample into a training set to train the model and a test set to evaluate it. Figure 4 shows the progress of both accuracy and loss in the training phase during 70 epochs. The accuracy is greater than 85% and the loss is less than 0.35.
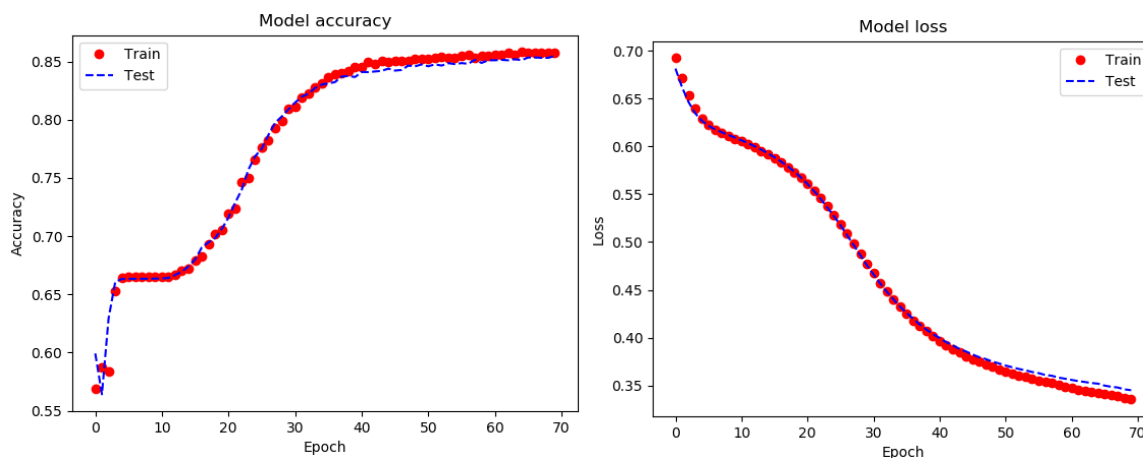

Figure 4. Training CNN accuracy and loss graphs for large feature configuration.

In the second experiment, another network's architecture is presented in Figure 5. This network works on small-feature configuration that includes frequency distribution, word-to-vector similarity and word emotion lexicon only. The network has four input layers. Each input layer represents an emotion space feature. The four input convolutional layers have 3 filters and window size 2, followed by an average pooling layer with pool size 3. There are 2 fully connected dense layers with sizes of 10 and 4 nodes, respectively.
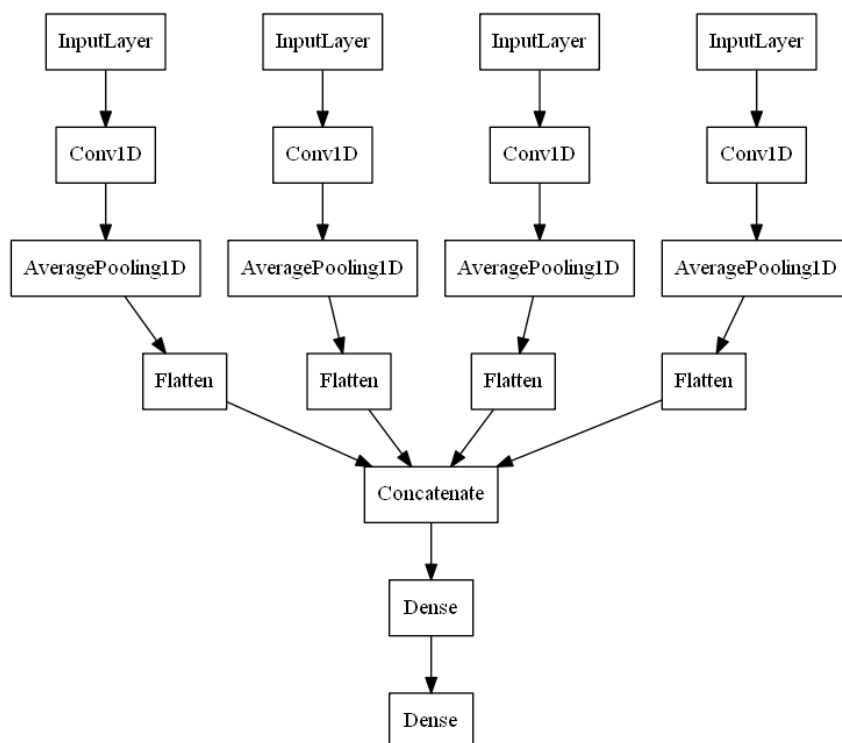
Figure 5. Small-feature CNN emotion classification architecture.

Figure 6 shows CNN model accuracy and loss graphs during the training phase. Figure 6 shows the progress of both accuracy and loss in the training phase in 70 epochs. The accuracy is greater than 85% and the loss is less than 0.35.

In these experiments, each convolutional layer has a four-or three-vector input according to feature set, Rectified Linear Units (ReLUs) [31] as activation functions and a batch size of 100. Filters are four

104

"Convolutional Neural Network Multi-Emotion Classifiers", S. S. Ibrahiem, S. S. Ismail, K. A. Bahnasy and M. M. Aref.
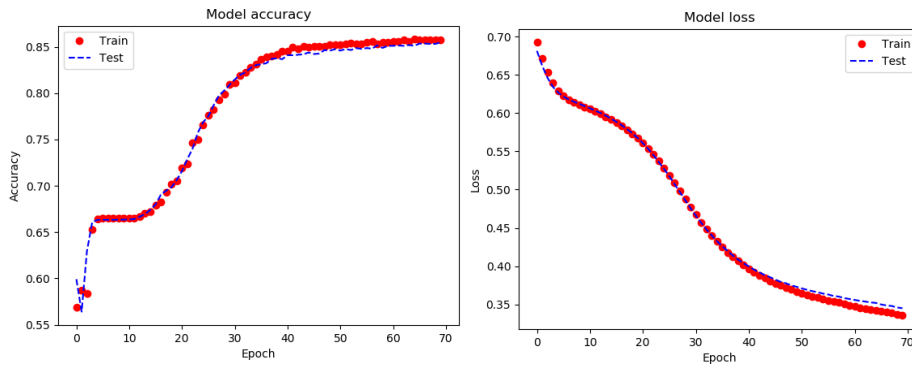
Figure 6. Training CNN accuracy and loss graphs for small-feature configuration.

and window is two, except the valence-arousal-dominance feature node with a three-vector input. Filters are three and window is two. The last decision layer is a sigmoid, so 0.5 is the threshold value to each output node. The model undergoes training through Adam optimizer [31] over shuffled mini-batches. The model stops the iterant processes of learning by a loss function binary cross-entropy [33]-[34].

4.2 Testing Models and Evaluation

The predefined training dataset consists of multi-labeled tweets. The output emotion classes are: anger, fear, joy and sadness. These tweets are trained on both convolutional neural networks in Figures 3 and 5. In the testing phase, the testing dataset consists of 1000 tweets from SemEval2018. They are multi-labeled in the same emotion range and never used in the training phase. Anger tweets are 392, fear tweets are 224, joy tweets are 402 and sadness tweets are 310. The proposed networks are evaluated by two different feature configurations with four standard machine learning algorithms; support vector machine, naïve Bayes, k nearest neighbour and multi-layer perceptron. Table 2 shows the precision (P), recall (R), f-measure (F) [35] values for each applied learning algorithm. Table 1 includes hamming score (HS), hamming loss (HL) and exact ratio (ER) [36] of the applied algorithms. First experiment evaluation has a mean of 0.756 and a standard deviation of 0.006. Second experiment evaluation has a mean of 0.745 and a standard deviation of 0.005.

Table 1. Hamming score, hamming loss and exact ratio on two feature configurations.

Techniques and feature configurations	HS	HL	ER
SVM			
Word2vec, FreqDist, NRC word lexicon	0.40	0.25	0.31
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.46	0.23	0.37
KNN			
Word2vec, FreqDist, NRC word lexicon	0.50	0.26	0.37
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.52	0.25	0.40
NB			
Word2vec, FreqDist, NRC word lexicon	0.38	0.28	0.26
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.43	0.27	0.30
MLP			
Word2vec, FreqDist, NRC word lexicon	0.40	0.27	0.28
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.44	0.25	0.31
Proposed CNN			
Word2vec, FreqDist, NRC word lexicon	0.46	0.24	0.35
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.52	0.23	0.42

On the other hand, Figure 7 shows another performance measurement for classification problem using Receiver Operating Characteristics (ROC) curve and Area Under the Receiver Operating Characteristics (AUC) [37]-[38]. Large-feature configuration network has 0.78 AUC, while small-feature one has 0.75 AUC.

By utilizing extensive computational power, convolutional neural network processing has been proven to be a very powerful method by researchers in many fields, like computer vision and natural language processing. Applied experiments contribute in multi-label classification in natural language processing field. They showed more reliable results and higher overall accuracies compared to standard machine learning algorithms.

Table 2. Precision, recall and f-measure results on two feature configurations.

	Anger			Fear			Joy			Sadness		
	P	R	F	P	R	F	P	R	F	P	R	F
SVM												
Word2vec, FreqDist, NRC word lexicon	0.68	0.42	0.52	0.71	0.18	0.29	0.76	0.58	0.66	0.56	0.33	0.41
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.74	0.46	0.57	0.77	0.23	0.35	0.81	0.67	0.73	0.61	0.34	0.44
K nearest neighbor												
Word2vec, FreqDist, NRC word lexicon	0.68	0.46	0.55	0.43	0.47	0.45	0.70	0.72	0.71	0.56	0.41	0.47
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.68	0.48	0.57	0.49	0.41	0.44	0.70	0.77	0.74	0.58	0.38	0.46
Naïve Bayes												
Word2vec, FreqDist, NRC word lexicon	0.61	0.45	0.51	0.44	0.31	0.37	0.75	0.57	0.65	0.46	0.38	0.42
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.63	0.46	0.53	0.46	0.42	0.44	0.78	0.63	0.70	0.50	0.44	0.47
MLP (three fully connected layers)												
Word2vec, FreqDist, NRC word lexicon	0.68	0.40	0.50	0.47	0.47	0.47	0.72	0.54	0.62	0.56	0.38	0.46
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.68	0.44	0.54	0.56	0.50	0.53	0.74	0.56	0.64	0.59	0.42	0.49
Proposed CNN												
Word2vec, FreqDist, NRC word lexicon	0.69	0.52	0.59	0.64	0.26	0.37	0.72	0.67	0.69	0.60	0.31	0.41
Word2vec, FreqDist, NRC word, hash, & VAD lexicons	0.66	0.64	0.65	0.86	0.08	0.15	0.76	0.74	0.75	0.64	0.38	0.47
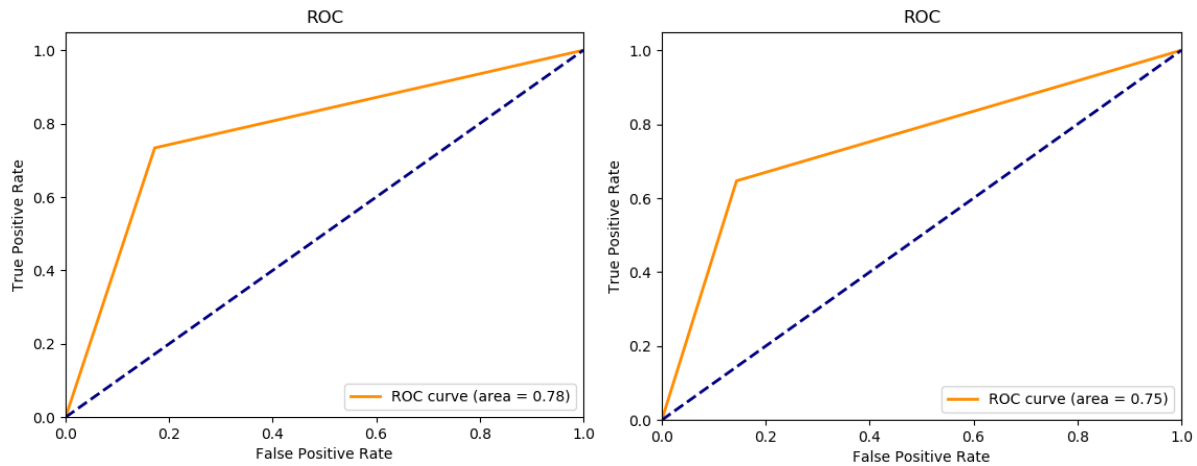
Figure 7. ROC curve of AUC calculation for large-feature configuration on the left-hand side and for small-feature configuration on the right-hand side.

5. CONCLUSION

This paper presented featured convolutional neural network architectures that applied multi-emotion classification in Twitter. It firstly discussed the related background about multi-label classification and emotion analysis. Secondly, it defined the annotated tweet datasets and lexicons that were used in pre-processing and feature extraction phases. Thirdly, it illustrated the architectures of the proposed convolutional neural networks and the applied experiments. Two experiments were applied using two different feature configurations. Fourthly, the evaluation metrics were illustrated to compare the CNN emotion classification models performance to the represented feature configurations and state-of-the-art classification algorithms. Python snapshots were shown to illustrate accuracy and loss performance during the training and testing phases. Finally, evaluation metrics were calculated and the proposed approach performance was evaluated. Tables 1 and 2 show the evaluation of the experimental results.

REFERENCES

[1]   X. Quan, Q. Wang, Y. Zhang, L. Si and L. Wenyi, "Latent Discriminative Models for Social Emotion Detection with Emotional Dependency," ACM Transactions on Information Systems (TOIS), vol. 34 no. 1, p. 2, 2014.

[2]   J. M. Nareshpalsingh and H. N. Modi, "Multi-label Classification Methods: A Comparative Study," International Research Journal of Engineering and Technology (IRJET), vol. 04, no. 12, December 2017.

[3]   C. N. N. Kamath, S. S. Bukhari and A. Dengel, "Comparative Study between Traditional Machine Learning and Deep Learning Approaches for Text Classification," Proc. of the ACM Symposium Conference, pp. 1-11, 2018.

[4]   M. Haggag, S. Fathy and N. Elhaggar, "Ontology-based Textual Emotion Detection," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 6, no. 9, pp. 239- 246, 2015.

[5]   Y. Cao, P. Zhang and A. Xiong, "Sentiment Analysis Based on Expanded Aspect-and Polarity-Ambiguous Word Lexicon," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 6, no. 2, 2015.

[6]   L. Flekova, E. Ruppert and D. P. Pietro, "Analyzing Domain Suitability of a Sentiment Lexicon by Identifying Distributionally Bipolar Words," Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2015), pp. 77–84, Lisboa, Portugal, September 2015.

[7]   D. M. El-Din, H. M. O. Mokhtar and O. Ismael, "Online Paper Review Analysis," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 6, no. 9, 2015.

[8]   S. Mohammad, E. Shutova and P. Turney, "Metaphor As a Medium for Emotion: An Empirical Study," Proceedings of the Joint Conference on Lexical and Computational Semantics, pp. 23-33, Berlin, Germany, 2016.

[9]   E. Cambria, J. Fu, F. Bisio and S. Poria, "AffectiveSpace 2: Enabling Affective Intuition for Concept-Level Sentiment Analysis," Proceedings of the 29th AAAI Conference on Artificial Intelligence, pp. 508-514, Austin, 2015.

[10]   Y. Wang and A. Pal, "Detecting Emotions in Social Media: A Constrained Optimization Approach," Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI), pp. 996-1002, Buenos Aires, Argentina, 2015.

[11]   P. Sobhani, S. M. Mohammad and S. Kiritchenko, "Detecting Stance in Tweets and Analyzing Its Interaction with Sentiment," Proceedings of the Joint Conference on Lexical and Computational Semantics, pp. 159–169, Berlin, Germany, August 2016.

[12]   N. Majumder, S. Poria, A. Gelbukh and E. Cambria, "Deep Learning-based Document Modeling for Personality Detection From Text," IEEE Intelligent Systems, vol. 32, no. 2, pp. 74-79, 2017.

[13]   R. Oramas, M. L. Barron-Estrada, R. Zatarain-Cabada and S. L. Ramírez-Ávila, "A Corpus for Sentiment Analysis and Emotion Recognition for a Learning Environment," Proc. of the 18th IEEE International Conference on Advanced Learning Technologies (ICALT), pp. 431-435, Mumbai, 2018.

[14]   M. Suhasini and S. Badugu, "Two Step Approach for Emotion Detection on Twitter Data," International Journal of Computer Applications, vol. 179, no. 53, pp. 12 –19, June 2018.

[15]   S. Mohammad, S. Kiritchenko, X. Zhu and J. Martin. "Sentiment, Emotion, Purpose and Style in Electoral Tweets," Information Processing and Management, vol. 51, no. 4, pp. 480–499, July 2015.

[16]   X. Sun, C. Sun, C. Quan, F. Ren, F. Tian and K. Wang, "Fine-grained Emotion Analysis Based on Mixed Model for Product Review," International Journal of Networked and Distributed Computing, vol. 5, no. 1, pp. 1–11, January 2017.

[17]   B. Gaind, V. Syal and S. Padgalwar, "Emotion Detection and Analysis on Social Media," Proceedings of the International Conference on Recent Trends in Computational Engineering and Technologies (ICTRCET'18), Bengaluru, India, May 2018.

[18]   S. Mohammad, F. B. Marquez, M. Salameh and S. Kiritchenko, "Semeval-2018: Affect in Tweets," Proceedings of the International Workshop on Semantic Evaluation (SemEval-2018), New Orleans, LA, USA, June 2018.

[19]   S. Mohammad and P. Turney, "Crowdsourcing a Word-Emotion Association Lexicon," Computational Intelligence, vol. 29, no. 3, pp. 436-465, 2013.

[20]   S. Mohammad and P. Turney, "Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon," Proceedings of the NAACL-HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, LA, California, June 2010.

[21]   S. Mohammad, "#Emotional Tweets," The 1st Joint Conference on Lexical and Computational Semantics, vol. 1 (Proceedings of the Main Conference and the Shared Task) and vol. 2 (Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012)), Montr'eal, Canada, pp. 246-255, 7-8 June 2012.

[22]   S. Mohammad, "Obtaining Reliable Human Ratings of Valence, Arousal and Dominance for 20,000 English Words," Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, July 2018.

[23]   V. A. Kharde and S. S. Sonawane, "Sentiment Analysis of Twitter Data: A Survey of Techniques," International Journal of Computer Applications, vol. 139, no. 11, pp. 5-15, April 2016.

[24]   T. Mikolov, G. Corrado, K. Chen and J. Dean, "Efficient Estimation of Word Representations in Vector Space," Proceedings of the International Conference on Learning Representations (ICLR 2013), pp. 1–12, 2013.

[25]   E. M. Alshari, A. Azman, S. Doraisamy, N. Mustapha and M. Alkeshr, "Improvement of Sentiment Analysis based on Clustering of Word2Vec Features," Proc. of the 28th International Workshop on Database and Expert System Applications, 2017.

[26]   K. K. Lurz, Natural Language Processing in Artificial Neural Network Sentence Analysis in Medical Papers, Master Thesis, Department of Astronomy and Theoretical Physics, Lund University, June 11, 2018.

[27]   F. Chollet, "keras," [Online], Available: GitHub. https://github.com/fchollet/keras, 2015.

[28]   Python Software Foundation, [Online], Available: http://www.python.org, 2019.

"Convolutional Neural Network Multi-Emotion Classifiers", S. S. Ibrahiem, S. S. Ismail, K. A. Bahnasy and M. M. Aref.

[29] J. Perkins, Python 3 Text Processing with NLTK 3 Cookbook, Packt Publishing, 2014.

[30] N. Hardeniya, "NLTK Essentials Build Cool NLP and Machine Learning Applications Using NLTK and Other Python Libraries," July 2015.

[31] A. F. Agarap, "Deep Learning Using Rectified Linear Units (ReLUs)," arXiv:1803.08375, 2018.

[32] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Proceedings of the 3rd International Conference on Learning Representations, 2014.

[33] S. Mannor, D. Peleg and R. Rubinstein, "The Cross-entropy Method for Classification," Proceedings of the 22nd International Conference on Machine Learning (ICML '05), pp. 561-568, Bonn, Germany, August 07 - 11, 2005.

[34] S. Baker and A. Korhonen, "Initializing Neural Networks for Hierarchical Multi-label Text Classification," Association for Computational Linguistics (BioNLP 2017), pp. 307-315, August 2017.

[35] D. Ganda and R. Buch, "A Survey on Multi-label Classification," Recent Trends in Programming Languages, vol. 5, no. 1, pp. 19-23, August 2018.

[36] S. R. Khade and S. R. Balwan. "Study and Analysis of Multi-label Classification Methods in Data Mining," International Journal of Computer Applications, vol. 159, no. 9, February 2017.

[37] J. A. Swets, "ROC Analysis Applied to the Evaluation of Medical Imaging Techniques," Invest. Radiol., vol. 14, no. 2, pp. 109-121, 1979.

[38] J. A. Hanley, "Receiver Operating Characteristic (ROC) Methodology: The State-of-the-Art," Crit. Rev. Diagn. Imaging, vol. 29, no. 3, pp. 307-335, 1989.

ملخص البحث:

تتسم اللغات الطبيعية بالكونية والمرونة، غير أنها لا يمكن أن توجد إلا أن تكون مصحوبة بشيء من الغموض. فوجود أكثر من منحنى وأكثر من معنى في سياق العبارة الواحدة هو السبب الأساسي في غموض الكلمات أو العبارات. لقد غطت غالبية الأبحاث السابقة المتعلقة بتحليل الأحاسيس التصنيف المبني على علامة مفردة فقط، في حين تم تجاهل وجود علامات إحساس متعددة في الكلمة أو العبارة ذاتها.

تعرض هذه الورقة تصنيفاً قائماً على تعدد الأحاسيس في نصوص تويتر (Twitter) يستخدم الشبكات العصبية الالتفافية. أما الخصائص المطبقة في هذا التصنيف فهي: معاجم الإحساس، ومضامين الكلمات، وتوزيعها التكراري. تم تقييم أداء الشبكات المقترحة عبر مقارنتها بأشهر خوارزميات التصنيف الشائعة. وأظهرت النتائج تفوق التقنية المقترحة عند تطبيقها على مجموعة من قواعد البيانات التي تمثّل تحدّياً في هذا المجال، وتراوحت نتيجة المبالغة بين 0.46 و 0.52.

FUZZY-ROUGH CLASSIFICATION FOR BRAINPRINT AUTHENTICATION

Siaw-Hong Liew[1], Yun-Huoy Choo[1] and Yin Fen Low[2]

ABSTRACT

The electroencephalogram (EEG) signal is used as biometric modality, because it is proven to be unique, universal and collectable. This work aims to assess the performance of fuzzy-based techniques for brainprint authentication modelling. We benchmark the performance of Fuzzy-Rough Nearest Neighbour (FRNN) technique to the Discernibility Nearest Neighbour (D-kNN) and the Fuzzy Lattice Reasoning (FLR) techniques using the selected samples of brainwaves' data from the original UCI EEG dataset. All the three classifiers are available in the fuzzy-rough version of WEKA implementation tool. Selected 9 EEG channels located at the midline and lateral regions were used in the experimentation. The coherence, mean of amplitudes and cross-correlation feature extraction methods were used to extract the EEG signals. The area under ROC curve (AUC) measurement of FRNN was promising against the D-kNN and FLR techniques. The FRNN model has achieved the best performance of AUC measure at 0.904 in opposition to the D-kNN and FLR models, where both recorded 0.770 and 0.563, respectively. However, the classification accuracy shows significantly no difference among the three classifiers. The results confirmed that the classification accuracy of D-kNN and FLR techniques is not reliable, because they are highly contributed by the true negative cases. Hence, we conclude that the FRNN model is less biased to imbalance data problem as compared to the D-kNN and FLR models. Future work of this research should focus on optimizing the EEG channel and feature selection in order to obtain a better data representation of biometric brainprint for more efficient authentication in imbalance data problem.

1. INTRODUCTION

The aim of brainprint authentication is to accept or reject the identity claimed by an individual. There are numerous types of person authentication methods, such as knowledge-based, token-based and biometric methods. The commonly used Personal Identification Number (PIN) and password are examples of knowledge-based authentication and signature is an example of token-based authentication. However, password and signature are considered the weakest authentication models, because the password can be stolen, while the signature can be forged easily. Biometric systems such as fingerprint, iris, face, voice and hand geometry authentication systems were introduced to overcome the security incompetency of the traditional authentication methods. Among all, fingerprint and face recognition are common modalities in today's biometric authentication systems. Fingerprint scheme [1] is widely used, but is still prone to forgery. This technology recognizes only the ridge arrangement on the finger surface, where intruders can easily replicate the fingerprint using silicon or gelatine to infringe the security systems. Facial recognition is also less promising, because the human face structure will change as a person ages. The above mentioned limitations can be overcome using a more secure biometric modality; the human brainprint. The brainprint extracted through electroencephalogram (EEG) signals is a highly secure biometric modality for person authentication. Over the recent years, EEG-based person authentication is catching much researchers' attention [2]-[3].

Various types of soft computing techniques have been applied in EEG signal classification. Artificial neural networks (ANNs), fuzzy logic, K-Nearest Neighbour (kNN), linear discriminant analysis (LDA) and support vector machine (SVM) are examples of soft computing techniques for EEG signal classification. Gui et al. [4] investigated visual evoked potential (VEP) data collection using a low-cost sensor system. ANNs were used for EEG-based biometric authentication and the classification accuracy achieved was around 90%. Back-propagation NN, SVM and LDA were used to classify the EEG signals

---

1. S. H. Liew & Y. H. Choo are with Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia. Emails: siawhong.liew@gmail.com and huoy@utem.edu.my
2. Y. F. Low is with Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia. Email: yinfen@utem.edu.my

for biometric authentication and the accuracy ranged from 80.8% to 89.5% [5]. Fuzzy logic is an outstanding model, because it can solve real-world classification problems, since there is no precise boundary existing between the categories of problems [6]. Furthermore, the fuzzy set theory is very flexible and allows simple computation of evidences in complex relations.

Due to that EEG signals are non-stationary, a combination of fuzzy set theory, rough set theory and kNN called Fuzzy-Rough Nearest Neighbour (FRNN) is chosen to classify EEG signals. It is because the uncertainty of fuzzy rough set is able to deal with this problem. The classification results will be compared with those of other fuzzy-rough approaches, such as Discernibility Nearest Neighbour (D-kNN) and Fuzzy Lattice Reasoning (FLR).

The rest of the paper is organized as follows: Section 2 provides a literature review on EEG signals for person authentication, feature extraction and soft computing techniques. Section 3 describes the classification techniques. Section 4 outlines the experimentation, which includes data pre-processing and preparation, feature extraction, experimental setting, performance measures and statistical tests. Section 5 depicts experimental results and discussion. Last but not least, Section 6 draws the conclusions and indicates the direction of future work.

2. RELATED WORK

Berger was the first who recorded the EEG signals in 1929 [7]. EEG is defined as the electrical activity recorded from the scalp surface [8]. EEG signals are the electromagnetic waves that are emitted from the human brain's neurons. EEG is the most practical capturing method that can be used in biometrics due to the advances in its hardware devices. The EEG recording is a completely non-invasive procedure that can be repeatedly applied to normal adults, patients and children with virtually no risk or limitation [8]. The main advantages of using brain electromagnetic waves are: the uniqueness and liveness of the EEG signals, in addition to that the recorded brain responses cannot be replicated and the individual's identity cannot be stolen. A research work [9] showed that the individual's EEG signals vary from every individual to another, even though they performed similar task or thought. Conditions of stress, anxiety, fatigue, medication, drowsiness, environment, …etc. can increase the difficulty of reproducing similar pattern of EEG signals [10]. For example, a person that has been under the influence of stress will generate different EEG signals when compared to his/her normal state.

EEG recording electrodes and their function are critical for obtaining high-quality data for interpretation [8]. One important problem of EEG signal recording is the artifacts. Examples of artifacts occurring in EEG signal recording are: eye blinking, head movements, muscle activities and electrocardiogram (ECG). Due to the very low amplitude of EEG signals, artifacts often contaminate the recordings, restricting or making difficult analysis or interpretation. Therefore, the position of the subject during EEG recording should be very comfortable to avoid unnecessary activities; a lying position diminishes the existence of some artifacts caused by feeble motion. One of the ideas that combined EEG signals with authentication systems was proposed by Thorpe et al. [11]. The studied authentication system was designed by using pass-thought, which is reliable due to the uniqueness of EEG signals. Apart from that, a consumer grade of EEG headset was used in Ashby et al. [12] for authentication purpose.

Marcel and Millán [13] achieved a high authentication performance of 93.4% in terms of accuracy. A total of 9 normal subjects were asked to perform 3 tasks (i.e., left-hand movement, right-hand movement and generation of words that begin with the same random letter) during 12 non-feedback sessions in 3 days, which means 4 sessions per day. The classification accuracy reached around 80% in the research work [14]. They analyzed the 8-channel EEG signals from a group of 40 volunteers who performed a simple experiment (i.e., relaxing with opened and closed eyes). In addition, the research work by Jian-Feng [15] used the BCI competition 2003 EEG dataset that was recorded from a total of 64 channels and sampled at 250 Hz. The authentication classification result ranged from 75% to 85%. Biometric authentication based on EEG signals conducted in [16] covered three tasks of classification accuracy of: reading task (97.3%), relax task (94.4%) and multiplication task (97.5%). The research work in [17] combined EEG headsets with the smartphone for EEG-based person authentication purpose. Besides, an EEG-based biometric authentication system was developed in [18]. The EMOTIV Epoch+ EEG headset was used to collect the EEG signals and the classification accuracy achieved was 96.97%.

Mean corresponds to the centre of a set of values. It is a time domain feature, which is calculated for the

111

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

reconstructed EEG signal amplitude and time duration. Mean has been used in [19] as one of the features for the filtered signals. The extracted signals are then distinguished to be normal or epileptic by using artificial neural network technique. Other than that, time domain features, such as mean, median, mode, standard deviation, minimum and maximum, were used in [20] in the analysis of EEG signals to detect brain abnormalities. Correlation is very similar to convolution, which is a mathematical operation. Then, cross-correlation is the measurement of the extent of similarity relationship between two signals. It is able to detect non-stationarity and is widely used for the analysis of time series of EEG signals. A set of five features is extracted and then utilized for training an SVM classifier to generalize the results. In this research, a healthy subject sample signals are acquired to represent a reference signal for data comparison. Cross-correlation has been aided to Support Vector Machine (SVM) classifier in EEG signal classification [21]. With the aid of cross-correlation, SVM is able to perform better in pattern recognition. The accuracy achieved was 94.5%. Hence, cross-correlation is a very useful technique to gain insight in EEG signals for feature extraction. Coherence is one of the feature extraction methods that is widely used for EEG signal analysis. Coherence is a linear correlation measure between two signals at different frequencies. It was first used as a feature in [22] for measuring the mean coupling between signals recorded from an electrode and its neighbours. In addition, mutual information, coherence and cross-correlation have been used in [23] for an EEG biometric system. The features extracted from the EEG signals have been proven unique enough among subjects for biometric applications. Research work in [23] used an unobtrusive authentication method that uses 2 frontal electrodes and 1 reference electrode placed at the left ear lobe only.

Due to low signal-to-noise ratio and non-stationarity of EEG signals, uncertainty modelling tools, such as fuzzy set and rough set, are needed to handle the related problems. Fuzzy set theory [24] and rough set theory [25] are good solutions in handling uncertainty and manipulating incomplete data. Fuzzy-rough set provides a higher degree of flexibility in dealing with imprecision and vagueness existing in real-world data [26]–[30]. Fuzzy-Rough Nearest Neighbour (FRNN) model introduced by Jensen and Cornelis [31] is hybridized with the strength of fuzzy-rough set and Fuzzy Nearest Neighbour (FNN) approach to complement each other. The constructed fuzzy lower and upper approximations are used to avoid the use of fuzzy logical connectives altogether. However, fuzzy-rough set allows that the element belongs to more than one class. In addition, FNN model is an extension version of kNN algorithm to fuzzy set theory and it is proved that FNN model outperforms the standard nearest neighbour model [32]. FNN model allows partial membership of an object in different classes and takes into account the closeness of each neighbour with respect to the test instance. Unfortunately, FNN algorithm is found out to have a problem when dealing with imperfect data. Therefore, the hybridization of the strength between fuzzy-rough set and FNN algorithm, which is fuzzy-rough nearest neighbour (FRNN) algorithm, can allow both to complement each other in order to gain good performance.

FRNN algorithm uses nearest neighbours to compute fuzzy lower and upper estimations in order to predict the test objects [31]. With the existing of the fuzzy approximations, the FRNN algorithm outperforms other nearest neighbour approaches and Naïve Bayes prediction models in classification problems. This is proven from the experiment done in Sarkar [33]. Three nearest neighbour approaches; namely, conventional kNN algorithm, the FNN algorithm and the FRNN algorithm, were used to classify Wisconsin Breast Cancer problem [34]. The dataset consisted of 699 samples and each sample provided ten numerical attributes. A total of 16 samples with missing attributes were removed from the dataset. From this experiment, FRNN algorithm gained the highest classification performance among the three algorithms. Moreover, the time complexity of FRNN algorithm is the same as those of the conventional kNN algorithm and the FNN algorithm. Furthermore, the FRNN algorithm was applied in China Stock Market Distressed Company for prediction problems [34]. The FRNN algorithm is able to use unbalanced and unmatched training and testing datasets in prediction. The prediction accuracy achieved was 78.37% which is better than that of the FNN classification approach. This study concluded that the FRNN approach performs better than the conventional kNN approach and the FNN approach. FRNN approach not only can deal with unbalanced data, but also performs well when dealing with incomplete data.

D-kNN approach is an extension of the kNN algorithm which uses the concept of discernibility. D-kNN computes the discernibility of the neighbours and the distances from the test objects. The main benefit in D-kNN approach is that it does not allow the classes of dataset to overlap. D-kNN algorithm considers the structural properties of the neighbours. A comparison of performance among three classifiers with

nearest neighbour approach was carried out by using Bupa Liver dataset [35]. The classifiers are conventional kNN algorithm, Weighted kNN (W-kNN) algorithm and D-kNN algorithm. By comparing the classifiers, D-kNN yields the best classification accuracy and net reliability. However, the processing time is slightly longer than in the conventional kNN and W-kNN algorithms. W-kNN yields the worse in terms of accuracy, because the dataset only contains six features and all of these are equally importance. W-kNN performed well when the dataset contains a larger number of features.

FLR is a rule-based classifier. The term "fuzzy lattice" was introduced by Naseem in 1994 [36] on the concept of fuzzy partial-order relation. The benefit of the lattice theory is capability of tackling with uncertain information and dealing with missing data [37]. Fuzzy lattices can be used in classification and clustering algorithms and have been successfully implemented in real-time problems, such as pattern recognition [38], air quality assessment and ambient ozone estimation [39]. FLR classifier was successfully applied in ambient ozone estimation [39] and the results with missing values and without missing values were compared. The FLR classifier gained similar values in terms of accuracy measure for the dataset with missing values and without missing values. The classification accuracy was 84.6% for dataset with missing values and 83.23% for dataset without missing values. Furthermore, the least time for training and testing was taken by FLR classifier. It used only around 1.5 seconds, while back-propagation neural networks took a training and testing time between 3 minutes and 25 minutes. In recent years, the FLR classifier has been used for image recognition, such as human facial expressions [40]-[41]. However, there is still lack of research on EEG signal classification using FLR.

Data pre-processing and feature extraction are the important steps in order to perform FLR classification. Seven different facial expressions; namely, neutral, angry, disgust, feared, happy, sad and surprised, were recorded. The dataset was divided into 75% of training data and 25% testing data. From this experiment, FLR classifier performed better than the conventional kNN algorithm [40].

3. CLASSIFICATION

In this study, FRNN, D-kNN and FLR techniques were used to accomplish brainprint authentication modelling. Brainprint authentication modeling consists only of 2 classes; client and impostor. The FRNN, D-kNN and FLR techniques can be found in fuzzy-rough version of Waikato Environment for Knowledge Analysis (WEKA). It is free downloaded from http://users.aber.ac.uk/rkj/book/wekafull.jar.

3.1 Fuzzy-Rough Nearest Neighbour (FRNN)

Fuzzy-Rough Nearest Neighbour (FRNN) was introduced by Jensen and Cornelis [31] in 2011. It is a hybrid model with the combination of fuzzy set, rough set and nearest neighbour classification approaches. In the FRNN algorithm, the lower and upper approximations are constructed by the nearest neighbours to allocate the decision class to the test object. The details of FRNN algorithm can be found in Algorithm 1 [31]. The FRNN algorithm calculates the similarity between the two objects and finally classifies the test objects into the most possible decision classes. FRNN classifies the test object based on single nearest neighbour with the highest similarity measure. Therefore, the value of k does not affect the classification performance. The FRNN technique captures uncertainty by using fuzzy-rough approximations. The construction of fuzzy upper and lower approximations is to avoid the use of fuzzy logical connectives completely. The connectives here are the keys in developing the fuzzy-rough set theory.

3.2 Discernibility Nearest Neighbour (D-kNN)

Discernibility Nearest Neighbour (D-kNN) classifier can handle overlapping classes of a dataset compared to the original kNN. The discernibility of the neighbours was first calculated, followed by their distances from the test objects. The algorithm of D-kNN is shown in Algorithm 2 [35]. The property of the neighbours is playing an important role in D-kNN prediction [35]. The ratio or distance of discernibility is computed for each neighbour data and the average of the ratios is taken for each class. D-kNN not only classifies the test elements based on the concept of nearest neighbours, but also based on the discernibility scores. The discernibility score of D-kNN classifier is produced for each object to be classified. After that, the average of the discernibility scores of the neighbouring objects and their distances from the objects are calculated for each one of the possible classes. Then, S_j is calculated for the classification score of each class. Eventually, the classification scores of the different classes are

compared in order to classify the test objects. The higher the classification score, the higher the chance to be the output of the classification.

Algorithm 1 Fuzzy-Rough Nearest Neighbour (FRNN) algorithm

1: **Input:** X, the training data; $\mathcal{C}$, the set of decision classes; y, the object to be classified

2: **Output:** Classification for y

3:   **begin**

4:     $N \leftarrow getNearestNeighbours(y, k)$

5:     $\tau \leftarrow 0, Class \leftarrow \emptyset$

6:     **foreach** $C \in \mathcal{C}$ **do**

7:       **if** $\left(\left(min_{a \in \mathbb{A}} \left(1 - \frac{|a(x)-a(y)|}{|a_{max}-a_{min}|}\right) \downarrow C \right)(y) + \left(min_{a \in \mathbb{A}} \left(1 - \frac{|a(x)-a(y)|}{|a_{max}-a_{min}|}\right) \uparrow C \right)(y) \right)/2 \geq \tau$

      then

8:       $Class \leftarrow C$

9:       $\tau \leftarrow \left(\left(min_{a \in \mathbb{A}} \left(1 - \frac{|a(x)-a(y)|}{|a_{max}-a_{min}|}\right) \downarrow C \right)(y) + \left(min_{a \in \mathbb{A}} \left(1 - \frac{|a(x)-a(y)|}{|a_{max}-a_{min}|}\right) \uparrow C \right)(y) \right)/2$

10:     **end**

11   **end**

12:    **output** $Class$

13:   **end**

Algorithm 2 Discernibility Nearest Neighbour (D-kNN) algorithm

---

1: **Input:** Input train objects (P), labels of train objects (T), number of neighbours (k), test objects (PT)

2: **Output:** Classification vector of test object (y)

3:   Initialization: set $i = 0$ (index of the objects), $j = 0$ (index over the classes)

4:   n $\leftarrow$ number of test objects (PT)

5:   N $\leftarrow$ number of train objects (P)

6:   z $\leftarrow$ discernibility vector for elements of P, using Algorithm-1 and P, T as inputs; $z = \{z_i\}$, $i = 1 \dots N$

7:   q $\leftarrow$ number of unique values of T (classes)

8:   do $i \leftarrow (i + 1)$

9:     **D** $\leftarrow$ vector of distances of PI (i) to P

10:     **sd** $\leftarrow$ sorted values of d

11:     **dk** $\leftarrow$ k first values of **sd**

12:     **v** $\leftarrow$ $\{v_m : v_m = z_m/dk_m\}, m = 1 \dots k$

13:     do $j \leftarrow (j + 1)$ for current i

14:     $C_j \leftarrow$ subset of k nearest elements of P belonging to the j-th class

15:     Classification score $S_j \leftarrow mean\left(v_{c_j}\right)$, $v_{c_j} = \{v_m : \forall m \in C_j\}$

16:   until $j = q$

17:     $b \leftarrow argmax S_j$

18:     $y_i \leftarrow$ class b

19:   until $i = n$

3.3 Fuzzy Lattice Reasoning (FLR)

Fuzzy Lattice Reasoning (FLR) is a classifier to extract rules from the input data based on fuzzy lattices. The sequence of the input data representation is important. FLR plays an important role in dealing with different types of data; for example, fuzzy sets, real vectors, images, symbols, graphs and waves. Other than that, FLR can deal with both points and intervals. Apart from that, FLR has the ability of knowledge representation and is capable of extracting implicit features beyond the data, which can represent the data as rules. Furthermore, FLR has the ability to combine different types of data, handle missing data and cope with both complete and incomplete lattices. FLR manages missing data by replacing them with least element O and great element I, respectively. For example, if the constituent lattice is $([0,1], \leq)$, then we can replace the missing data by intervals of $O = [1,0]$ and $I = [0,1]$, respectively [37].

Algorithm 3 Rule Induction by the Fuzzy Lattice Reasoning (FLR) algorithm

---

1: A rule base $RB = \{A_1 \rightarrow C_1, \dots, A_L \rightarrow C_L\}$ is given – Note that RB could initially be empty, i.e. $L = 0$.

2: Present the next input rule $a_i \rightarrow c_i, i = 1, \dots, n$ to the initially 'set' family of rules in RB.

3: If no more rules in RB are 'set' then

4:     Store input rule $a_i \rightarrow c_i$ in RB;

5:     $L \leftarrow L + 1$;

6:     Goto 2,

7: Else, compute the fuzzy degree of inclusion $k(a_i \leq A_l), l \in \{1, \dots, L\}$ of antecedent a_i to the antecedents of all the 'set' rules in RB

8: Competition among the 'set' rules in RB.

9:     Winner is rule $A_J \rightarrow C_J$ such that $J = \arg \max_{l \in \{1,\dots,L\}} k(a_i \leq A_l)$.

10: The *Assimilation Condition*: Both $diag_1(a_i \vee A_J)$ is less than a maximum user-defined threshold size D_{crit} and $c_i = C_J$.

11: If the *Assimilation Condition* is satisfied then

12:    Replace the antecedent A_J of the winner rule $A_J \rightarrow C_J$ by the join-interval $a_i \vee A_J$.

13: Else, 'reset' the winner rule $A_J \rightarrow C_J$.

14: Goto 3,

Algorithm 4 Generalization by the Fuzzy Lattice Reasoning (FLR) algorithm

---

1: Consider a rule base $RB = \{A_1 \rightarrow C_1, \dots, A_L \rightarrow C_L\}$.

2: Present a rule antecedent 'a_0' for classification based on the rule base RB.

3: Calculate the fuzzy degree of inclusion $k(a_0 \leq A_l), l \in \{1, \dots, L\}$ of antecedent 'a_0' in the antecedents 'A_l', $l = 1, \dots, L$ of all rules in RB.

4: Competition among the rules in RB.

5:     Winner is rule $A_J \rightarrow C_J$ such that $J = \arg \max_{l \in \{1,\dots,L\}} k(a_i \leq A_l)$.

6: The antecedent a_0 is classified to the class with label C_J.

(a_i, C_L) is the representation for the input datum to the FLR model, where C_L represents the class label of datum a_i and can be interpreted as a rule "if a_i then C_L". An input datum (a_0, C_0) is presented to the network in the learning phase. The degree of inclusion between input and stored rules in RB will be calculated as $k(a_0, a_1), \dots, k(a_0, a_c)$. The FLR will choose the rule with $\arg \max_{l \in \{1,\dots,L\}} k(a_i \leq A_l)$ as the winner rule. If the winner rule A_J and input datum a_0 have the same class label and the size of $a_i \vee A_J$ is less than a user-defined threshold, then the winner rule will be updated. There is only one parameter that can be tuned in FLR; that is the threshold size, D_{crit}. D_{crit} is used to indicate the maximum size of a

115

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

hyperbox to be learned. Larger values of D_{crit} will result in more generalized rules while smaller values of D_{crit} will result in more specific rules.

4. EXPERIMENTATION

EEG signal classification is a difficult task as a result of that the characteristics of EEG signals are non-stationary, in addition to high dimensionality and low signal-to-noise ratio (SNR). Thus, data pre-processing and data preparation steps are important.

4.1 Data Pre-processing and Data Preparation

In this study, a free EEG dataset is used which can be taken from UCI Machine Learning Repository [42]. The online available EEG dataset consists of three versions, which are small dataset, large dataset and full dataset. Each of the datasets contains an individual, 10 individuals and 122 individuals, respectively. The UCI EEG dataset was recorded for both alcoholic and non-alcoholic persons. Since this study focused on person authentication modeling, only non-alcoholic dataset will be used. Alcoholic data is not suitable for this study, because data collected from alcoholic persons might be less accurate due to their brains having been affected by alcohol. Large dataset will be used in this study, but one of the individuals from the large dataset will be replaced by an individual from the full dataset. This is because there are many redundant trials by the individuals which will affect the result. Each individual accomplished 60 trials. This EEG dataset consists of the measurements of 64 electrodes (61 active electrodes + 3 reference electrodes) placed on the scalps and the sampling rate was at 256Hz.

The stimuli were composed of 90 images that were chosen from a total of 260 black-and-white Snodgrass and Vanderwart image set [43]. The subjects were requested to recognize the image as soon as the image is displayed on the computer screen. The distance of the computer screen from the subject's eyes was 1 meter. The image remained on the screen for 300ms and the Inter-Stimulus Interval (ISI) for each test was set to 3200ms. The visual stimulus presentation is illustrated in Figure 1.
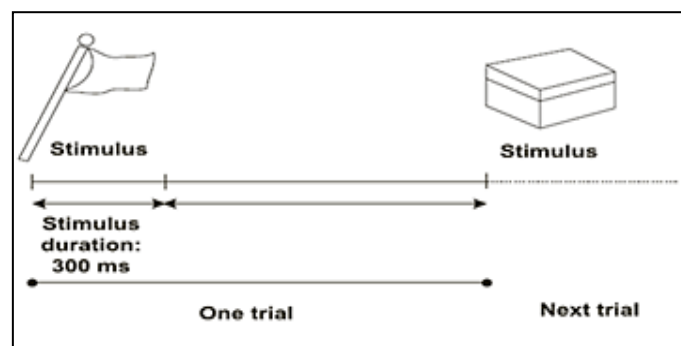
Figure 1. Visual stimulus presentation.

In the general machine learning model building, there are some common suggestions for train/test splits, such as 60/40, 70/30, 80/20 or even 90/10, if the dataset is relatively large [34]. The higher percentage of train data tends to generate a better model, but sacrifices the objectivity of test results due to low number of test data. Therefore, the larger the dataset, the higher the train/test proportion which may be applied. However, machine learning experiment seldom implements the 90/10 proportion, unless the dataset used is extremely large. We used the 80/20 train/test proportion in this study, where 480 instances were used for model building *versus* 120 instances for model testing.

In this stage, we have checked the trials in order to avoid the redundant trials between train set and test set. The dataset has equivalent distribution of trials between S1 object, S2 match and not-match for both train set and test set. In this study, we only selected 100 data points, which corresponds to approximately 300 milliseconds (ms). This is because the VEP normally occurred within the first 300ms. Besides, the EEG signals of S2 were different from those of S1 due to that S2 involves brain information about the match or not-match analysis. Only the electrodes located at the midline and lateral sides were considered. This is due to that midline and lateral electrodes provide stronger strength from electrical signals when responding to visual stimuli [44]. The lateral electrodes are PO7, PO8, O1, O2 and OZ, while the midline electrodes are FPZ, FZ, CZ, PZ and OZ.

4.2 Feature Extraction

A set of feature vectors were retrieved from the raw EEG dataset. The extracted feature vectors act as a different observation for the purpose of classification. Besides, feature extraction can reduce the dimensions of the input attributes as compared to the raw EEG dataset. In this study, coherence, cross-correlation and mean of amplitudes are selected from a particular literature review. The three feature extraction methods are described as follows:

a) Coherence: Coherence is used in order to compute the degree of linear correlation between two signals. The correlation between two signals at different operating frequencies can be revealed by coherence [17]. EEG-based coherence analysis is proven to be suitable for use in biometrics [45]. Coherence is ranging from 0 to 1, where the value of 0 indicates that the two signals are independent, while the value of 1 indicates that the two signals are completely linearly dependent. The coherence is calculated as follows:

$$C_{xy}(f) = \frac{|P_{xy}(f)|^2}{P_{xx}(f)P_{yy}(f)} \tag{1}$$

where,

$C_{xy}(f)$ is a function of the power spectral density (P_{xx} and P_{yy}) of x and y and the cross-power spectral density (P_{xy}) of x and y.

b) Cross-correlation: Cross-correlation, known as a sliding dot product, is used to compute the similarity between two signals. It is also frequently used to obtain the existence of a known signal sequence in an unknown one. It is a function of the relative delay between the signals and the application in pattern recognition. Two input signals will be calculated for cross-correlation:

Channel 1 with itself: ρ_X,

Channel 2 with itself: ρ_Y,

Channel 1 with channel 2: ρ_{XY}.

The correlation ρ_{XY} between two random variables x and y with expected values μ_X and μ_Y and standard deviation σ_X and σ_Y is given as:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\big((X - \mu_X)(Y - \mu_Y)\big)}{\sigma_X \sigma_Y} \tag{2}$$

where, $E(\ )$ is the expectation operator and $cov(\ )$ is the covariance operator.

c) Mean of Amplitudes: Mean, also known as average, is the sum the of all EEG potential values divided by the number of data points. The mean is calculated as follows:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \tag{3}$$

where, n is the number of data points and x_i is the value of the data.

4.3 Experimental Setting

In FRNN algorithm, fuzzy logic connectives are crucial for developing the fuzzy-rough set theory. A triangular norm (t-norm), $\mathcal{T}$ is any increasing, associative and commutative $[0,1]^2 \to [0,1]$ mapping satisfying $\mathcal{T}(1, x) = x$, for all x in $[0,1]$. On the contrary, an implicator is any $[0,1]^2 \to [0,1]$ mapping ℓ satisfying $\ell(0,0) = 1, \ell(1,x) = x$, for all x in $[0,1]$. Based on the [31], the Kleene-Dienes implicator for x, y value in $[0,1]$ was implemented. In addition, the experimental setting for D-kNN was the same with FRNN classifier. Kleene-Dienes was chosen for both t-norm and implicator. Moreover, there is only one parameter that can be tuned in FLR algorithm; that is the threshold size ρ. We have set $\rho = 0.1$ [38] in our experiment.

4.4 Performance Measures and Statistical Test

The experimental result is analyzed based on the accuracy and the area under the receiver operating characteristics (ROC) curve (AUC). The AUC measure is used as one of the performance measures in

117

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

this study, because it is more reliable and statistically consistent as compared to the accuracy measure in [46]. The accuracy and AUC of FRNN will be compared with the results obtained from D-kNN and FLR. The purpose of this comparison is to test whether FRNN can perform better than other classification algorithms, such as D-kNN and FLR.

Beforehand, the normality distribution of data is verified by using the Anderson-Darling test. The Anderson-Darling test [47] is modified from Kolmogorov-Smirnov (K-S) test. By comparing to the (K-S) test, Anderson-Darling test contributes more weights to the distribution tails. The critical value is calculated for the specific distribution. The Anderson-Darling test is calculated as:

$$W_n^2 = n \int_{-\infty}^{\infty} [F_n(x) - F^*(x)]^2 \psi\, (F^*(x) dF^*(x) \tag{4}$$

where, ψ = non-negative weight function which can be defined from:

$$\psi = F^*(x)(1 - F^*(x))^{-1} \tag{5}$$

The normality distribution of data must be determined before performing a statistical test. A statistical test is performed in order to determine the confidence level of the dataset which lead to reaching conclusions. Parametric test is chosen when data is normally distributed, while non-parametric test will be chosen when data is not normally distributed. Parametric tests, such as Z-test, paired-sample t-test or F-test, will yield higher accuracy when data is normally distributed. Simultaneously, if data is normally distributed and a non-parametric test is performed, then the results will not be as accurate as in the case of parametric test.

From the normality test using Anderson-Darling test, the accuracy of FRNN, accuracy of D-kNN and AUC of D-kNN are normally distributed, while the accuracy of FLR, AUC of FRNN and AUC of FLR are not normally distributed. Therefore, a paired-sample t-test is performed between accuracy of FRNN and accuracy of D-kNN. In contrast, Wilcoxon signed-rank test is performed when the results are not normally distributed.

A paired-sample t-test is performed to compare the differences of means between paired observations by using the IBM SPSS Statistics 22. The paired-sample t-test is a statistical validation method which is used to compare the means from different sources in a dataset [48]. The reason behind this is to investigate the significance differences between two groups. The null hypothesis of paired-sample T-test states that the difference between two mean values is zero, which is represented as:

$$H_0 : \mu_1 - \mu_2 = 0 \tag{6}$$

On the other hand, Wilcoxon signed-rank test is frequently used for non-parametric testing. It is an alternative method for paired-sample t-test. The Wilcoxon signed-rank test is used to evaluate the difference of medians between paired data. Wilcoxon signed-rank test is more powerful in distinguishing the differences between two samples [49]. The nominal data cannot be analyzed with Wilcoxon signed-rank test, because the difference of the nominal data points has no specific value.

In statistical test, the null hypothesis is rejected if and only if the p-value is less than 0.05, which means that there are statistically significant differences between the two samples. On the contrary, the null hypothesis is accepted if and only if the p-value is larger than 0.05, which means that there are no statistically significant differences between the two samples. A statistical test was performed to test the differences between the two classifiers for different cases of use at 95% confidence level.

5. EXPERIMENTAL RESULTS AND DISCUSSION

Table 1 shows the individual EEG signals' authentication performance for 10 subjects. D-kNN model ranked first with an accuracy of 92.09%, while the FLR model came second with an accuracy of 90.67%. Meanwhile, the FRNN model had the lowest classification accuracy, recorded to be 90.17%, as compared to the D-kNN model and the FLR model. The highest accuracy recorded for FRNN model was up to 99.17%, while the lowest accuracy was 80.83% only. On the other hand, the accuracy of D-kNN model was considerably high, where the highest value recorded was 98.33% and the lowest was 89.17%. Besides, the accuracy of FLR model was between 88.33% and 94.17%. In overall, the accuracy of FRNN model, D-kNN model and FLR model indicated good classification results.

Table 1. Comparison of accuracy and AUC between FRNN, D-kNN and FLR.

Subject	FRNN		D-kNN		FLR	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
Subject 1	87.50	0.924	91.67	0.735	92.50	0.773
Subject 2	86.67	0.788	89.17	0.661	90.83	0.579
Subject 3	88.33	0.922	91.67	0.758	90.00	0.500
Subject 4	80.83	0.704	89.17	0.488	88.33	0.491
Subject 5	93.33	0.954	94.17	0.880	90.00	0.500
Subject 6	88.33	0.924	90.00	0.814	90.00	0.500
Subject 7	99.17	1.000	98.33	1.000	94.17	0.708
Subject 8	90.83	0.895	91.67	0.733	90.00	0.500
Subject 9	90.00	0.936	90.83	0.758	90.00	0.500
Subject 10	96.67	0.990	94.17	0.875	90.83	0.579
Average	**90.17**	**0.904**	**92.09**	**0.770**	**90.67**	**0.563**

The FRNN model has achieved the highest average AUC with 0.904. However, the average AUC has been found to be 0.770 in D-kNN model followed by FLR model with 0.563. Thus, it was shown that the FRNN model outperformed the D-kNN model and the FLR model. In the FRNN model, the highest AUC achieved was a perfect 1.00 score, while the lowest was 0.704. Comparatively, the highest AUC for D-kNN model also achieved the perfect score 1, but 0.488 was the lowest AUC in this model. Lastly, 0.773 was the highest AUC in FLR model, while 0.491 was the lowest AUC in FLR model.

From the performance measure in terms of accuracy, D-kNN model is slightly higher than FRNN and FLR models. On the other hand, FLR model showed the worst result in AUC. From Table 1, we can visually observe that most of the results are around 0.5, which is rated as worst classification performance. The model is not capable of distinguishing between positive class and negative class. This result explained that the FLR model is not suitable to classify EEG signals for person authentication. The possible reason is the parameter setting of FLR model. As previously described, there is only one parameter which can be tuned; that is the size of threshold. As the parameter setting for FLR model is 0.1 [38], therefore it will affect the AUC obtained, since the perspective of AUC is different from that of accuracy. The larger values of threshold will result in more generalized rules [39]. The threshold used in this project is small, which is 0.1; therefore, the rules are more specific as smaller values of threshold will result in more specific rules. Thus, the AUC of FLR models is lower than in FRNN and D-kNN models.

By observing the overall classification results above, FRNN gained good performance in terms of accuracy and AUC compared to D-kNN and FLR models. As previously described, FRNN algorithm is a fusion model that combines the strength fuzzy-rough set and the FNN approach. The decision class is determined by using the fuzzy lower and upper approximations to compute the membership value of a test object [31]. The fuzzy lower and upper approximations play a crucial role in dealing with noisy data such as EEG signals. Hence, the FRNN is able to perform better.

Table 2. Statistical test for comparison of accuracy and AUC between FRNN, D-kNN and FLR.

Performance Measure	Mean	p - Value	Statistical Test
FRNN Accuracy	90.17	0.071	No significant differences
D-kNN Accuracy	92.09		
FRNN Accuracy	90.17	0.767	No significant differences
FLR Accuracy	90.67		
FRNN AUC	0.904	0.004	Significant differences
D-kNN AUC	0.770		
FRNN AUC	0.904	0.006	Significant differences
FLR AUC	0.563		

119

Table 2 shows the statistical test for the comparison of accuracy and AUC among FRNN, D-kNN and FLR models. The paired-sample t-test was only used for the comparison between the accuracy of FRNN model and the accuracy of D-kNN model. The p-value of this comparison is 0.071, which is greater than 0.05. Thus, we can conclude that the FRNN model and the D-kNN model are not significantly different. On the contrary, Wilcoxon signed-rank test was used for the rest of the comparisons. The p-value for the comparison between the accuracy of FRNN model and that of FLR model was recorded at 0.767, which is greater than 0.05. Hence, there are also no significant differences between the FRNN model and the FLR model. In summary, accuracy comparisons did not show significant differences among the models.

Comparatively, the statistical test for the comparison between the AUC of FRNN model and D-kNN model showed significant differences with a p-value of 0.004. From the mean values in Table 2, it is clearly proved that the AUC of FRNN model is higher than the AUC of D-kNN model. Thus, we can conclude that the FRNN model performed better than the D-kNN model. Furthermore, a statistical test was also carried out for the comparison between the AUC of FRNN model and that of FLR model. The p-value was recorded at 0.006 and indicating significant differences in the paired set. The AUC of FRNN model achieved a value of 0.904, while the AUC of FLR model achieved only 0.563, which is considered a poor result. In other words, the FRNN model significantly performed better than the D-kNN model and the FLR model.

6. CONCLUSIONS

Among the fuzzy set and rough set approaches, the FRNN model is proven to be significantly better than D-kNN model and FLR model in EEG signal classification for brainprint authentication modeling. The AUC of FRNN model is 0.904, which is considered an excellent classification result. However, further work should be done on the FRNN model to improve the accuracy and AUC, since a good authentication system should have a perfect classification. The classification results gained from the FRNN model are more stable and consistent as compared to the classification results of D-kNN model and FLR model. This study showed the importance and capability of fuzzy-rough approximations of handling uncertain and non-stationary signals.

ACKNOWLEDGEMENTS

REFERENCES

[1] P. Rajeswari, S. V. Raju, A. S. Ashour and N. Dey, "Multi-fingerprint Unimodel-based Biometric Authentication Supporting Cloud Computing," Intelligent Techniques in Signal Processing for Multimedia Security, Springer, Cham, pp. 469–485, 2017.

[2] S. Yang and F. Deravi, "On the Usability of Electroencephalographic Signals for Biometric Recognition: A Survey," IEEE Trans. Human-Machine Syst., vol. 47, no. 6, pp. 958–969, 2017.

[3] K. P. Thomas and A. P. Vinod, "Toward EEG-based Biometric Systems: The Great Potential of Brain-Wave-based Biometrics," IEEE Systems, Man, & Cybernetics Magazine, pp. 6–15, 2017.

[4] Q. Gui, Z. Jin and W. Xu, "Exploring EEG-based Biometrics for User Identification and Authentication," Proc. of the IEEE Signal Processing in Medicine and Biology Symposium (SPMB), pp. 1–6, 2014.

[5] H. U. Jian-Feng, "Comparison of Different Classifiers for Biometric System Based on EEG Signals," Proc. of the 2nd International Conference on Information Technology and Computer Science, pp. 288–291, 2010.

[6] O. F. Bay and A. B. Usakli, "Survey of Fuzzy Logic Applications in Brain-related Researches," Journal of Medical Systems, vol. 27, no. 2, pp. 215–223, 2003.

[7] H. Berger, "Ueber das Elektroencephalogramm des Menschen," J. für Psychol. und Neurol., vol. 87, pp. 527–570, 1929.

[8] M. Teplan, "Fundamentals of EEG Measurement," Measurem. Sci. Review, vol. 2, no. 2, pp. 1–11, 2002.

[9] A. Zuquete, B. Quintela and J. P. Silva Cunha, "Biometric Authentication using Brain Responses to Visual

Stimuli," Proc. of the International Conference on Bio-inspired Sys. and Sig. Proces., pp. 103–112, 2010.

[10] I. Svogor and T. Kisasondi, "Two-factor Authentication Using EEG Augmented Passwords," Proc. of the 34th International Conference on Information Technology Interfaces, pp. 373–378, 2012.

[11] J. Thorpe and P. C. Van Oorschot, "Pass-thoughts : Authenticating with Our Minds," Proceedings of the 2005 Workshop on New Security Paradigms, pp. 45–56, 2005.

[12] C. Ashby, A. Bhatia, F. Tenore and J. Vogelstein, "Low-cost Electroencephalogram (EEG) Based Authentication," Proc. of the 5th Int. IEEE/EMBS Conf. on Neural Engineering, pp. 442–445, 2011.

[13] S. Marcel and J. D. R. Millán, "Person Authentication Using Brainwaves (EEG) and Maximum *A Posteriori* Model Adaptation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 4, pp. 743–752, Apr. 2007.

[14] R. B. Paranjape, J. Mahovsky, L. Benedicenti and Z. Koles', "The Electroencephalogram As a Biometric," Proc. of the Canadian Conference on Electrical and Computer Engineering, vol. 2, pp. 1363–1366, 2001.

[15] J.-F. Hu, "Multi-feature Biometric System Based on EEG Signals," Proc. of the 2nd Int. Conf. on Interaction Sciences Information Technology, Culture and Human (ICIS '09), pp. 1341–1345, 2009.

[16] C. R. Hema, M. P. Paulraj and H. Kaur, "Brain Signatures: A Modality for Biometric Authentication," Proc. of the International Conference on Electronic Design, pp. 1–4, 2008.

[17] H. Olesen, J. Klonovs and C. K. Petersen, Development of a Mobile EEG-based Feature Extraction and Classification System for Biometric Authentication, Master Thesis, Aalborg Univ., Copenhagen, 2012.

[18] I. Jayarathne, M. Cohen and S. Amarakeerthi, "BrainID : Development of an EEG-based Biometric Authentication System," Proc. of the 7th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016.

[19] S. Garg and R. Narvey, "Denoising and Feature Extraction of EEG Signals Using Wavelet Transform," International Journal of Engineering, Science and Technology (IJEST), vol. 5, no. 6, pp. 1249–1253, 2013.

[20] M. Kalaivani and V. A. Devi, "Analysis of EEG Signal for the Detection of Brain Abnormalities," Proceedings of the International Conference on Simulations in Computing Nexus (IJCA), pp. 1–6, 2014.

[21] S. Chandaka, A. Chatterjee and S. Munshi, "Cross-correlation Aided Support Vector Machine Classifier for Classification of EEG Signals," Expert Systems with Applications Journal, vol. 36, no. 2, pp. 1329–1336, Mar. 2009.

[22] E. Gysels and P. Celka, "Phase Synchronization for the Recognition of Mental Tasks in a Brain-Computer Interface," IEEE Trans. Neural Syst. Rehabil. Eng., vol. 12, no. 4, pp. 406–415, 2004.

[23] A. Riera, A. Soria-Frisch, M. Caparrini, C. Grau and G. Ruffini, "Unobtrusive Biometric System Based on Electroencephalogram Analysis," EURASIP J. Adv. Signal Process., vol. 2008, no. 1, pp. 1–8, 2008.

[24] L. Zadeh, "Fuzzy Sets," Information and Control Journal, vol. 8, no. 3, pp. 338–353, 1965.

[25] P. Zdzislaw, "Rough Sets," International Journal of International Conference on Computer and Applications (IJCIS), vol. 11, no. 5, pp. 341–356, 1982.

[26] R. Jensen and Q. Shen, "New Approaches to Fuzzy-rough Feature Selection," IEEE Trans. Fuzzy Syst., vol. 17, no. 4, pp. 824–838, 2009.

[27] R. Sadeghi and J. Hamidzadeh, "Automatic Support Vector Data Description," Soft Computing Jour., pp. 147–158, 2016.

[28] M. Zabihimayvan, R. Sadeghi, H. N. Rude and D. Doran, "A Soft Computing Approach for Benign and Malicious Web Robot Detection," Expert Systems with Applications, vol. 87, pp. 129–140, 2017.

[29] J. Hamidzadeh, M. Zabihimayvan and R. Sadeghi, "Detection of Website Visitors Based on Fuzzy Rough Sets," Soft Computing, vol. 22, no. 7, pp. 2175–2188, 2018.

[30] M. Zabihimayvan and D. Doran, "Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection," Proc. of the IEEE International Conference on Fuzzy Systems, 2019.

[31] R. Jensen and C. Cornelis, "Fuzzy-rough Nearest Neighbour Classification and Prediction," Theor. Comput. Sci., vol. 412, no. 42, pp. 5871–5884, Sep. 2011.

[32] J. M. Keller and M. R. Gray, "A Fuzzy K-nearest Neighbor Algorithm," IEEE Trans. Syst. Man. Cybern., vol. 15, no. 4, pp. 580–585, 1985.

[33] N. Mac Parthaláin and R. Jensen, "Fuzzy-rough Approaches for Mammographic Risk Analysis," Intell. Data Anal., vol. 14, no. 2, pp. 225–244, 2010.

[34] S. Raschka, "Model Evaluation, Model Selection and Algorithm Selection in Machine Learning," University of Wisconsin-Madison, arXiv.org > cs > arXiv:1811.12808, 2018.

[35] Z. N. Voulgaris, Discernibility Concept in Classification Problems, PhD Thesis, Uni. of London, 2009.

[36] N. Ajmal and K. Thomas, "Fuzzy Lattices," Inf. Sci. (NY), vol. 79, no. 3–4, pp. 271–291, 1994.

[37] V. Petridis and V. G. Kaburlasos, "Fuzzy Lattice Neural Network (FLNN): A Hybrid Model for Learning," IEEE Trans. Neural Networks, vol. 9, no. 5, pp. 877–890, 1998.

[38] Y. Jamshidi Khezeli and H. Nezamabadi-Pour, "Fuzzy Lattice Reasoning for Pattern Classification Using a New Positive Valuation Function," Adv. Fuzzy Syst., vol. 2012, pp. 1–8, 2012.

[39] V. G. Kaburlasos, I. N. Athanasiadis and P. A. Mitkas, "Fuzzy Lattice Reasoning (FLR) Classifier and Its Application for Ambient Ozone Estimation," Int. J. Approx. Reason., vol. 45, pp. 152–188, 2007.

[40] S. E. Papadakis, V. G. Kaburlasos and G. A. Papakostas, "Fuzzy Lattice Reasoning (FLR) Classifier for Human Facial Expression Recognition," World Scientific Proceedings Series on Computer Engineering and Information Science, pp. 633–638, 2012.

[41] S. E. Papadakis, V. G. Kaburlasos and G. A. Papakostas, "Two Fuzzy Lattice Reasoning (FLR) Classifiers and Their Application for Human Facial Expression Recognition," Jou. Mult. Log. Soft Comput., vol. 22, no. 4–6, pp. 561–579, 2014.

[42] H. Begleiter, "EEG Database," UCI Machine Learning Repository, [Online], Available at: https://archive.ics.uci.edu/ml/datasets/eeg+database.

[43] J. G. Snodgrass and M. Vanderwart, "A Standardized Set of 260 Pictures: Norms for Name Agreement, Image Agreement, Familiarity and Visual Complexity," Jou. Exp. Psychol. Hum. Learn. Mem., vol. 6, no. 2, pp. 174–215, Mar. 1980.

[44] J. V. Odom et al., "ISCEV Standard for Clinical Visual Evoked Potentials (2009 Update)," Doc. Ophthalmol., vol. 120, no. 1, pp. 111–119, Feb. 2010.

[45] V. Marsalek, P. Matousek, M. Mautner, Merta and R. Moucek, "Coherence of EEG Signals and Biometric Signals of Handwriting under Influence of Nicotine, Alcohol and Light Drugs," Neural Netw. World, vol. 16, pp. 41–60, 2006.

[46] C. X. Ling, J. Huang and H. Zhang, "AUC : A Better Measure than Accuracy in Comparing Learning Algorithms," Springer Verlag, Berlin, Heidelberg, pp. 329–341, 2003.

[47] T. W. Anderson, "Anderson-Darling Tests of Goodness-of-Fit," International Encyclopedia of Statistical Science, no. 2. Springer Verlag, Berlin, Heidelberg, pp. 52–54, 2011.

[48] A. Field, Discovering Statistics Using SPSS, 3rd Edition, Sage Publications, 2009.

[49] C. A. Bellera, M. Julien and J. A. Hanley, "Normal Approximations to the Distributions of the Wilcoxon Statistics: Accurate to What N? Graphical Insights," Jou. Stat. Educ., vol. 18, no. 2, pp. 1–17, 2010.

ملخص البحث:

تُستخدم إشارات التخطيط الكهربائي للدّماغ كوسيلة قياس بيولوجية، لأنها تتمتع بالفرادة والعمومية وقابلية الجمع. يهدف هذا العمل الى تقييم أداء تقنيات قائمة على المنطق المشوَّش من أجل نمذجة إثبات أصالة الصور باستخدام بصمات أدمغة الأشخاص. استخدمت مجموعة بيانات (EEG UCI) الأصلية للمقارنة بين ثلاثٍ من تقنيات التصنيف القائمة على المنطق المشوّش، تم اختيار تسع قنوات تحتوي على إشارات التخطيط الكهربائي للدّماغ وتتموضع عند خط الوسط والمناطق الجانبية لاستخدامها في إجراء التجارب. ويقترح الباحثون أن يركز البحث المستقبلي على اختيار قناة التخطيط الكهربائي للدّماغ والسِّمات المتعلقة بها من أجل الحصول على تمثيلٍ أفضل للبيانات الخاصة ببصمات أدمغة الأشخاص بهدف التوصل الى قدرٍ أكبر من الدقة في إثبات أصالة صور وجوه الأشخاص للتغلب على مشكلة عدم التوازن.

LOW PASS AND QUAD BAND PASS TUNABLE FILTER BASED ON STUB RESONATOR TECHNIQUE

Yanal S. Faouri, Hanin Sharif, Leena Smadi and Hani Jamleh

ABSTRACT

An electronically tunable multi-passband filter using one varactor diode is implemented based on transmission line stub method for passing several most favourable applications through multiple operating bands. In this paper, the filter is designed on Rogers RT/Duroid 5880-substrate and its input and output ports are terminated by 50 Ω microstrip feed line. The filter passbands consist of low pass filter (LPF) with tunable cut-off frequency which can reach 0.94 GHz, then several tunable bandpass filters (BPFs) that can cover the following frequency ranges BPF1 (1.94 – 3.33 GHz), BPF2 (3.83 – 4.23 GHz), BPF3 (4.53 – 5.56 GHz) and BPF4 (6.83 – 7.48 GHz) with insertion loss (IL) of $|S_{21}| \leq 3$ dB. The designed filter is the binomial type with 3 elements that are implemented in three shunt stubs with the middle stub being shorted. A parametric study was conducted for the optimum location of the varactor diode and an external DC biasing circuit introduced to produce the required reverse biasing for the varactor diode and its effect was considered. The demonstrated filter is investigated using the high-frequency structure simulator (HFSS). The measured scattering parameters' S_{11} (reflection coefficient) and S_{21} (transmission coefficient) results show good agreement with the simulated values.

KEYWORDS

1. INTRODUCTION

Modern microwave circuits and high-frequency applications are utilizing electronically tunable filters in order to cover more bandwidth and to be able to support multiple services. This will reduce the number of circuits used for different applications by one circuit that can support many applications and hence the complexity and system cost are reduced. In order to meet the increasing demands on rejecting out-of-band noise and jamming spectral components while simultaneously supporting multiple information channels, new developments in the design of tunable microwave filters are necessary [1]. Micro-strip filters are preferred due to their attractive features, like simplicity in manufacturing, reduction in cost, being easily integrated in circuits, high speed and high data rate capability.

Filters are tunable in many ways, including mechanical, magnetic and electronic methods. Mechanical filters are considered the best regarding power handling capability, linearity and quality factor. They have found few applications due to their huge size, low speed and large weight. Filters based on magnetic methods, like ferrimagnetic resonance, magnetostatic wave, evanescent waveguide, E-plane printed circuit and yttrium iron garnet (YIG) for providing multiple tunable bands, are preferred for higher selectivity and smaller size. On the other hand, electronic methods are done by employing semiconductors, such as PIN or varactor diodes, which can provide larger bandwidth with low IL, high selectivity, fast tuning, higher stability and lower DC biasing [2]. In contrast, micro-electromechanical systems (MEMSs) have indeed lower size, but they suffer from poor quality factor and need a large DC biasing voltage. Using MEMSs in tunable filters has been widely reported in the literature. Three-channel filter bank with two MEMS switches operates in the range 14 – 20 GHz, where each channel has a fixed three-pole end-coupled bandpass filter and the results showed an insertion loss between 1.7 and 2 dB as given in [3]. Instead of using a capacitor bank for filter tuning, it can be achieved by varying the resonator capacitive loading [4]-[5]. A single bandpass filter can also be tuned by changing the capacitive loading and hence adjusting the resonator physical size directly [6], where the filter can be tuned in the frequency range from 12 to 15 GHz with IL better than 3 dB. MEMS switches have been used as on-off elements between the resonators and the extra feed lines to form an interdigitated coplanar bandpass filter [7]; the filter can cover the range 18.5 – 21 GHz with IL less than 3.5 dB. Filters with

Y. S. Faouri*, H. Sharif, L. Smadi and H. Jamleh are with Electrical Engineering Department, Faculty of Engineering, University of Jordan, Amman, Jordan. Email: *y.faouri@ju.edu.jo

tunable characteristics employing a magnetic material such as YIG or gadolinium gallium garnet (GGG) have been utilized to design a wideband bandstop filter. Varying the magnetic field can tune the absorption occurrence and hence the center frequency rejection in the range 2.5 – 23 GHz using a microstrip line with GaAs substrate [8].

Employing varactor diodes for an electronically reconfigurable filter is another common method. A dual bandpass filter implemented using stepped impedance resonators has been reported and the center frequency tuning range for the two bands was from 0.7 – 1.2 GHz with a 3-dB fractional bandwidth (FBW) of 11.29 – 14.77 % (FBW $= 2 \times \frac{f_H - f_L}{f_h + f_L} \times 100\%$) and 1.4 – 2.15 GHz with a 3-dB FBW of 8 – 9.38 %, respectively [9]. A three-stage dual-band BPF using varactor diode and implemented using a stepped impedance method has been reported in [10] to be tuned from 1.75 – 2.66 GHz and 3.85 – 4.54 GHz with 70 MHz average bandwidth. A high-selectivity tunable dual-band bandpass filter implemented using varactor loaded resonators [11] had its first band center frequency ranging from 570 – 700 MHz with tunable 3-dB bandwidth changing from 36 – 60 MHz, while the other passband can be tuned from 1.156 – 1.336 GHz with the bandwidth varying from 67 – 85 MHz. The measured IL was from 2.3 – 2.03 dB. One tunable microstrip bandpass filter with center frequency ranging from 380 – 920 MHz and having its 3-dB bandwidth varying from 27 – 38 MHz has been reported in [12] through utilizing two varactors. Chebyshev stepped impedance filter with three shorted stubs and three varactors have been used to produce one tunable passband from 1.09 – 2.44 GHz with a reverse biased voltage changing from 0 – 30 V [13]. Combline dual bandpass filter implemented on microstrip substrate and tuned by using four varactor diodes was used to shift the center frequency from 1.85 to 1.95 GHz and from 2.1 to 2.2 GHz for the two bands, respectively [14]. The filter 3-dB bandwidth was less than 3.2 % with a tuning range of 250 MHz. Three varactors were utilized to tune the BPF lower and upper edges from 760 – 840 MHz and from 981 – 1107 MHz, respectively. The FBW was tuned from 15.5 – 36 % as reported in [15]. A BPF design using ring resonator technique with coupled feed lines was reported in [16] by using four PIN diodes as switching elements to alternate between narrowband filter and wideband filter. The 3-dB FBW could be tuned from 58.5 – 75 % at 2.4 GHz center frequency. A filter with lowpass and bandpass bands has been proposed in [17] based on three shunt stubs and loaded with a varactor diode. The LPF cut-off frequency could be tuned from 0.61 – 0.93 GHz while the bandpass was tuned from 1.85 – 3.27 GHz.

In this paper, a novel tunable planar line microwave filter design is proposed and investigated. It consists of five tunable bands; an LPF band and four BPF bands. These bands are tuned by means of varactor diode. The filter schematic and dimensions are outlined in Section 2. The proposed filter consists of three shunt stubs with varactor being connected to the middle stub for tuning the LPF cut-off frequency and the center frequency for the BPF bands. The prototype filter characteristics and the design procedure are described in Section 3. Results of simulation and discussion are presented in Section 4. The experimental verifications and a comparison with the simulated results are outlined in Section 5. Finally, the conclusion is given in Section 6.

2. FILTER STRUCTURE

The proposed filter consists of three shunt stubs with 50 Ω input and output feed lines; the first and third stubs are open-circuited, while the second stub is short-circuited. The low passband and quad bandpass regions are created with acceptable impedance bandwidth and IL at the passband and good S_{11} values at the pass frequencies by adjusting the length and width of the three stubs, the location of the varactor and the DC biasing circuit element value. The geometrical parameters for the proposed filter are shown in Figure 1, where all dimensions are carefully obtained by parametric analysis in order to achieve the desired response over the needed frequency range. The overall filter dimensions are 5.1×10.8 cm^2. It is fabricated on a double-side Rogers RT/Duroid 5880 substrate with tan δ = 0.0009 and $\varepsilon_r = 2.2$ with 0.8 mm thickness. The supply positive and negative terminals are connected to the varactor cathode and anode, respectively. The footprint for the inductor and resistor has been taken as 1.2×0.6 mm^2, while for the varactor, it is 1×0.8 mm^2 and the other filter parameters are listed in Table 1.

Table 1. Dimensions for the proposed filter geometry in mm.

W1 = 23	W2 = 62	W3 = 1.2	W4 = 12	Substrate width = 51
L1 = 2.5	L2 = 1.2	L3 = 44.125	L4 = 22	Substrate length = 108

124

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.
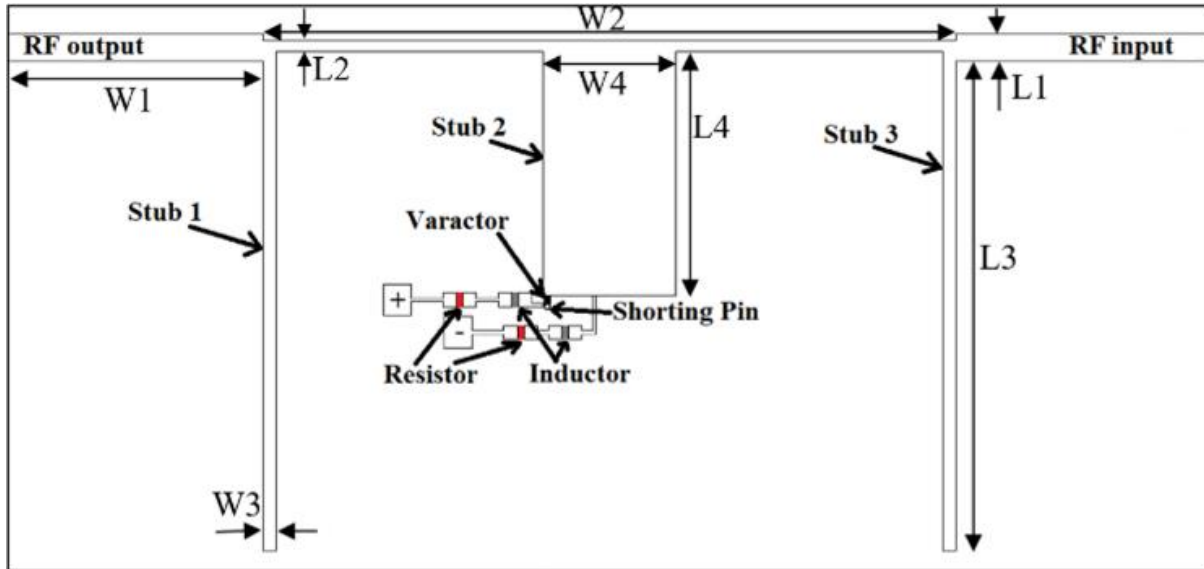
Figure 1. The proposed filter geometry with two open-circuited stubs and one short-circited stub.

3. DESIGN PROCEDURE

The design starts with a maximally flat LPF prototype consisting of three elements based on the IL method. The lumped circuit elements of the low-pass prototype filter are found to be $g_o = 1, g_1 = 1, g_2 = 2, g_3 = 1, g_4 = 1$. The prototype filter is modified to act as a bandpass filter by transforming every element of the LPF to a combination of an inductor and capacitor [18].

The bandpass filter with lumped elements is implemented using a microstrip transmission line model, which is done using the Kuroda Identities (first and second kinds) and Richards' transformations (to convert the lumped elements into transmission lines). The resulting stub dimensions have been calculated based on the chosen substrate and considering the open or short termination of the transmission line stubs that act as resonators.

The varactor type SMV1232, which is a hyper-abrupt junction, tunes the varactor from Skyworks [19] that works under a reverse biasing voltage varying between $0 - 15$ V and can provide an appropriate quality factor 'Q' in wireless systems for frequencies over 10 GHz. It is being inserted to change the filter overall capacitance and hence the resonant frequency of the band will change accordingly.

4. RESULTS AND ANALYSIS

The proposed filter was simulated using HFSS and several parametric studies have been conducted to achieve a filter with multi-pass bands which cover many highly demanded services. The filter first band covers low frequencies and its cut-off frequency can be tuned from 710 to 940 MHz. It can be used to pass LTE 700, GSM 3G 800, GSM 3G 900, industrial, scientific and medical (ISM) applications. At 200 MHz and beyond from the cut-off frequency, the attenuation can reach 50 dB.

The second passing filter is a bandpass filter BPF1 which is tuned from $1.94 - 3.33$ GHz with FBW ranging from $2.74 - 8.4$ % as the biasing voltage decreases and it has an average BW of 155 MHz while being tuned. This band can be utilized to pass many services, such as; wireless fidelity (Wi-Fi), Bluetooth, GSM 2100 and GSM 2300. This band has low IL varying from $0.38 - 0.98$ dB. The third passband BPF2 covers frequencies from $3.83 - 4.23$ GHz with FBW $1.55 - 9.93$ % for a reverse biasing voltage form $3 - 15$ V with an IL ≤ 3 dB, whereas below 3 V, the filter has a relatively higher loss that can reach 7 dB. The tuning BW for this passband ranged from 60 to 400 MHz with an average of 263 MHz. This band can be used for worldwide interoperability for microwave access (Wi-MAX) applications. The fourth band BPF3 ranges from $4.62 - 5.56$ GHz with an average BW of 397 MHZ and it is suitable for C-band uplink and downlink and Wi-Fi applications and its FBW can be tuned from $5.54 - 15.38$ %. The last passing band BPF4 has an average BW of 366 MHz and can cover frequencies from $6.85 - 7.43$ GHz, which makes it suitable for satellite communications with FBW tuned from 6.64 $- 8.12$ %. The cut-off frequency for the LPF and the upper 'f_H' and lower 'f_L' frequencies for the four

" Low Pass and Quad Band Pass Tunable Filter Based on Stub Resonator Technique", Y. S. Faouri, H. Sharif, L. Smadi and H. Jamleh.

passbands are listed in Table 2. Some cases for which the IL ≥ 3 dB in the BPF2 band are identified in Table 2.

Table 2. Proposed filter passing frequencies for all varactor capacitance values.

Varactor	LPF	BPF 1		BPF 2		BPF 3		BPF 4	
C (pF)	f_C (GHz)	f_L (GHz)	f_H (GHz)	f_L (GHz)	f_H (GHz)	f_L (GHz)	f_H (GHz)	f_L (GHz)	f_H (GHz)
4.15	0.71	1.94	2.11			4.62	5.39	6.85	7.43
3.22	0.80	2.07	2.28			4.53	5.18	6.97	7.28
2.67	0.84	2.18	2.43	IL is higher than 3dB		4.47	4.97	6.91	7.34
2.28	0.87	2.29	2.54			4.71	5.38	6.74	7.40
1.97	0.89	2.41	2.66			4.74	5.37	6.88	7.48
1.72	0.89	2.54	2.78			4.41	4.90	6.92	7.36
1.51	0.91	2.65	2.87	3.83	3.89	4.79	5.38	6.83	7.40
1.35	0.91	2.76	2.96	3.83	3.92	4.80	5.35	6.96	7.24
1.22	0.92	2.86	3.03	3.83	3.95	4.85	5.36	6.91	7.22
1.13	0.92	2.93	3.09	3.83	3.99	4.93	5.43	6.83	7.27
1.05	0.93	2.99	3.14	3.83	4.02	4.96	5.42	6.92	7.27
0.99	0.93	3.04	3.18	3.83	4.05	5.05	5.48	6.91	7.29
0.94	0.93	3.08	3.21	3.82	4.06	5.07	5.48	6.94	7.27
0.90	0.93	3.10	3.23	3.83	4.09	5.02	5.38	6.93	7.29
0.86	0.93	3.14	3.25	3.83	4.11	5.06	5.43	6.99	7.20
0.84	0.93	3.16	3.27	3.83	4.12	5.11	5.48	6.91	7.27
0.81	0.93	3.17	3.28	3.83	4.15	5.09	5.42	6.93	7.21
0.78	0.94	3.21	3.31	3.83	4.18	5.20	5.54	7.01	7.31
0.76	0.94	3.22	3.32	3.83	4.19	5.23	5.54	6.93	7.34
0.75	0.94	3.22	3.32	3.83	4.21	5.22	5.54	6.84	7.33
0.74	0.94	3.23	3.32	3.83	4.21	5.26	5.55	6.96	7.34
0.73	0.94	3.24	3.33	3.83	4.21	5.22	5.53	6.99	7.30
0.72	0.94	3.24	3.33	3.83	4.23	5.26	5.56	6.84	7.31

The required coupling parameters are the external quality factor (Q_e) and the mutual coupling and they are displayed in Figure 2 for BPF1 and BPF3. It can be shown that the proposed filter has high Q_e and lower mutual coupling. External quality factor describes the amplitude at resonance and is calculated using Equation 1, while mutual coupling determines the spacing required between adjacent elements and is calculated using Equation 2.

$$Q_{ei}^{(n)} = \frac{g_i g_{i+1}}{(FBW)_n} \tag{1}$$

$$M_{i,i+1}^{(n)} = \frac{(FBW)_n}{\sqrt{g_i g_{i+1}}} \tag{2}$$

where FBW is defined earlier in Section 1, 'n' is the number of the pass band (n = 1, 2, 3, 4) and 'i' is

126

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

an index for the LPF prototype element (i = 0, 1, 2, 3, 4) given in Section 3. From the equations above, $Q_{e1}^{(n)} = Q_{e4}^{(n)}$, $Q_{e2}^{(n)} = Q_{e3}^{(n)}$ and $M_{12}^{(n)} = M_{23}^{(n)}$.
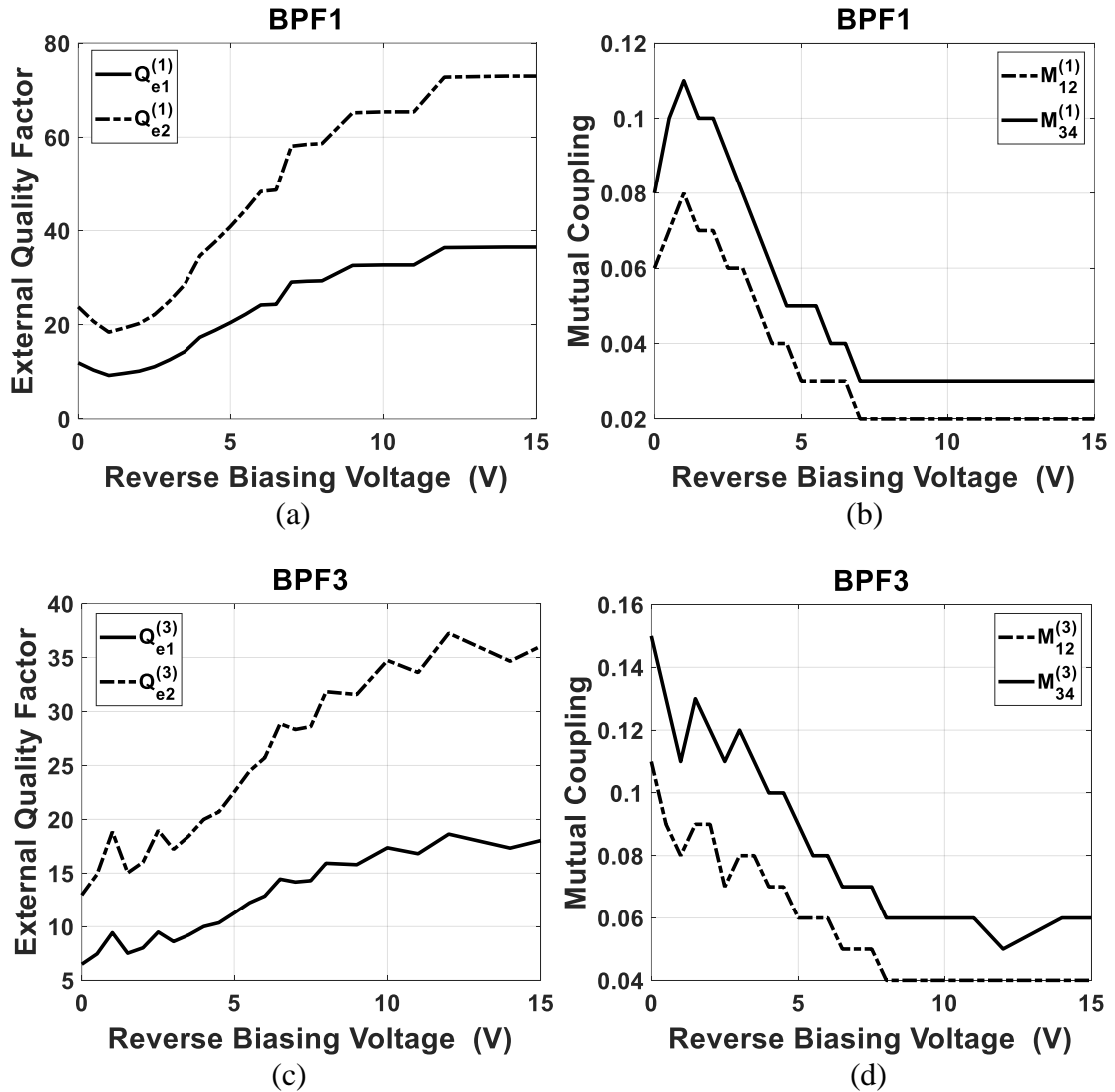
Figure 2. Filter parameters (a) external quality factor for BPF1 (b) Mutual coupling for BPF1, (c) external quality factor for BPF3 and (d) Mutual coupling for BPF3.

Scattering parameters S_{21} and S_{11} are the most important filter characteristics which display the utilized or rejected frequencies. The previously explained bands can be noticed from S_{21} and S_{11} curves shown in Figure 3 and Figure 4, respectively for selected varactor capacitances. The return loss (RL) within the passband is greater than 15 dB for all tuning states.

It can be noticed from Figure 3 that the filter has two fixed transmission zeros between BPF1 and BPF2 bands and between BPF3 and BPF4 bands, while the filter has two tunable transmission zeros (TZs) located between the LPF and BPF1 bands and between the BPF2 and BPF3 bands.

The current distribution shown in Figure 5 demonstrates the part of the filter which is active at the selected frequency through monitoring the filter element that has the highest surface current which is 50 A/m. At a frequency of 10 MHz, which is in the LPF band, the input and the output and the transmission line connecting the stubs are responsible for this mode of operation as shown in Figure 5(a). For a frequency of 1.23 GHz, which belongs to the rejected band, it can be noticed that most of the filter has no current passing through, since the signal was inserted from the right port to the left port and was reflected and dissipated through the first stub as can be noticed from Figure 5(b). At 2.14 GHz, which is the center frequency for the bandpass filter mainly, the first stub is responsible for this band named BPF1 as shown in Figure 5(c). For the second passband named BPF2, the current distribution is shown in Figure 5(d), where the middle and third stubs are active at 4 GHz. At a frequency of 4.49 GHz, which

belongs to the BPF3, the filter middle stub is responsible as shown in Figure 5(e) and for the BPF4 at 7.15 GHz, the whole filter except the shorted stub has higher maximum surface current with higher order modes being transmitted in the transmission lines as shown in Figure 5(f).
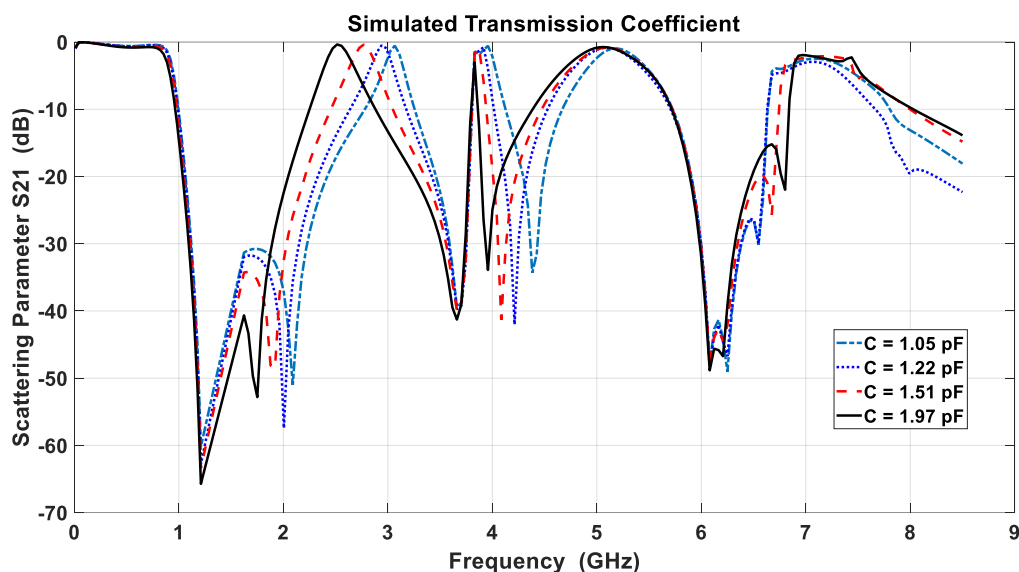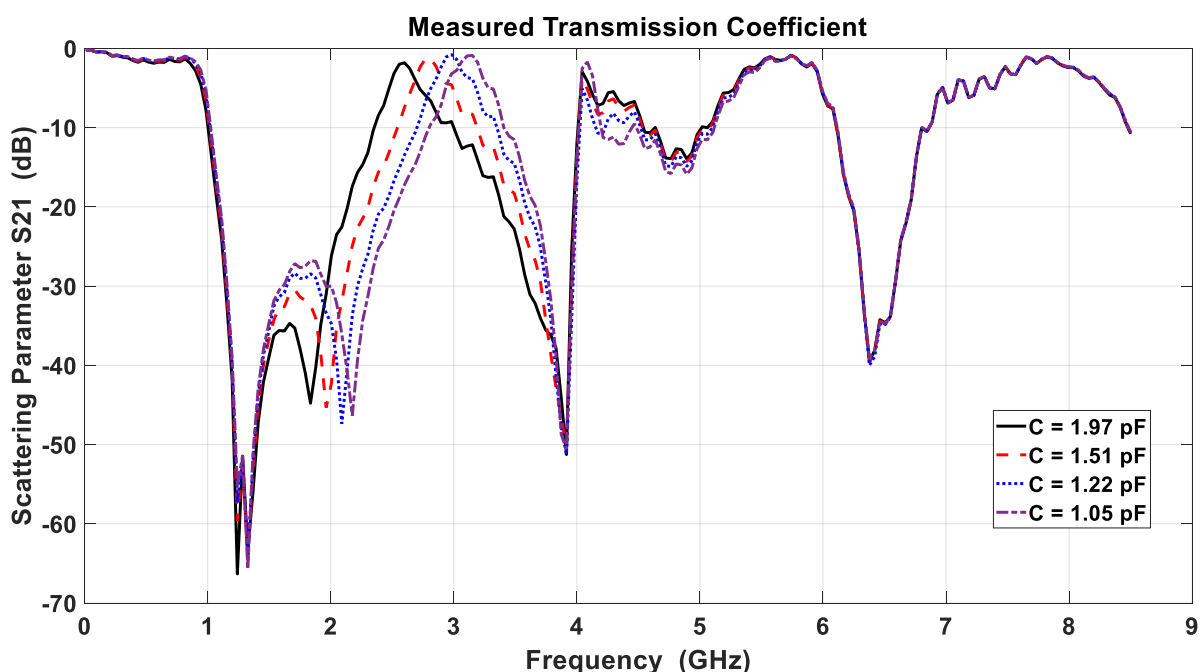

Figure 3. Simulated transmission coefficients for various varactor capacitances.
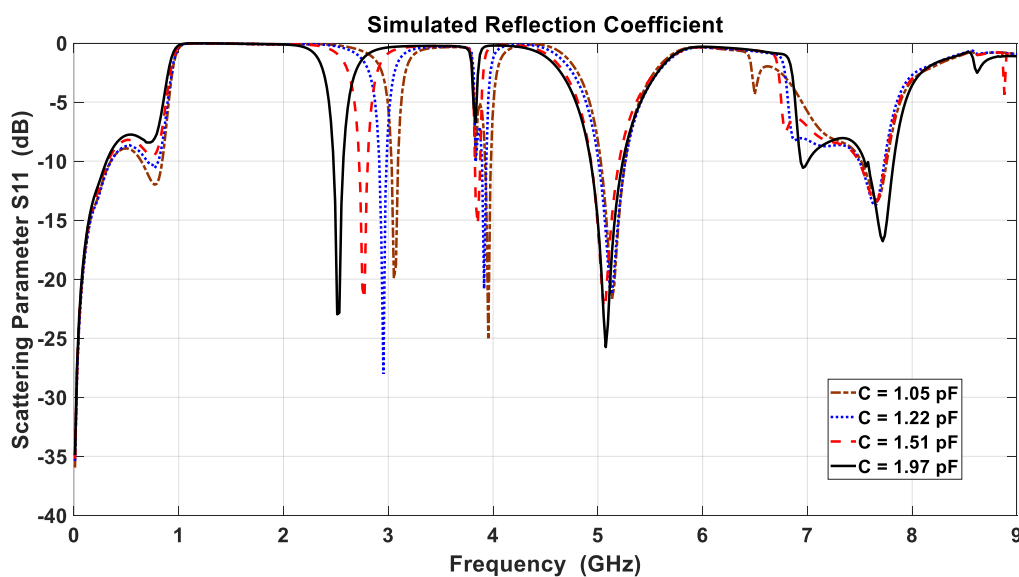
Figure 4. Simulated reflection coefficients for various varactor capacitances.
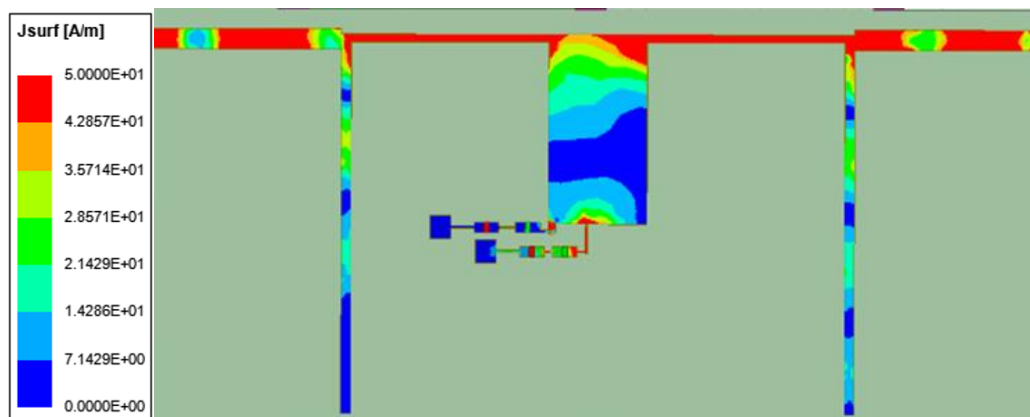
(a)
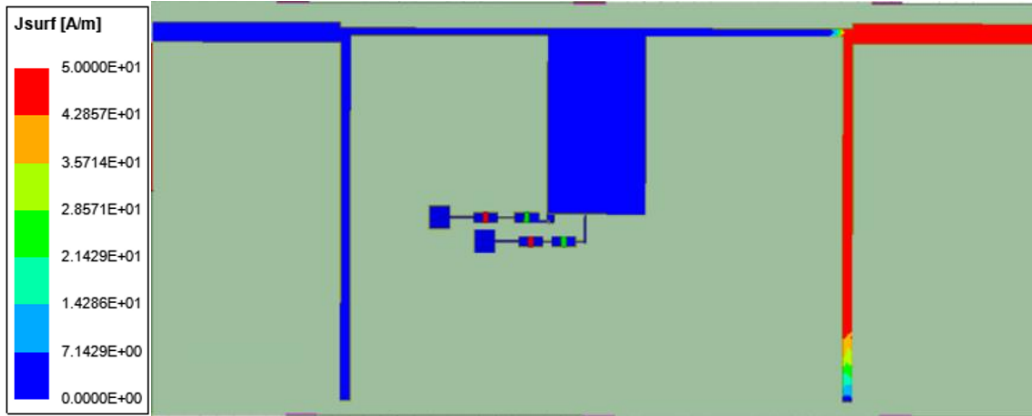
(b)
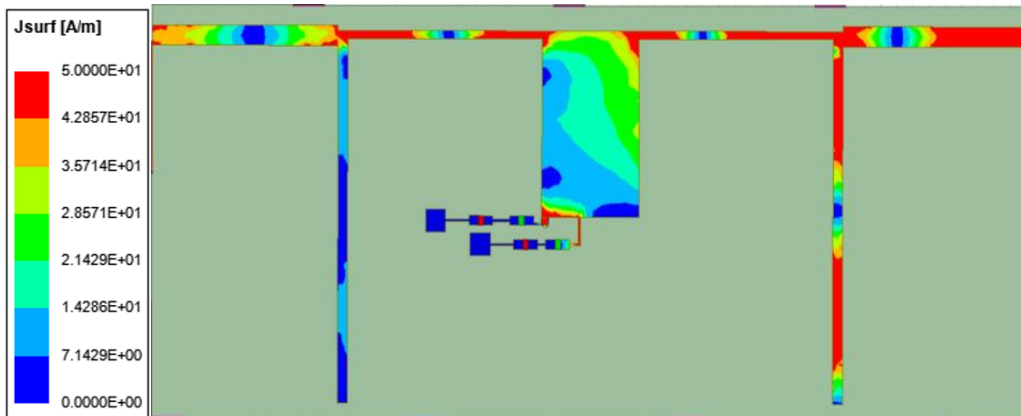
(c)
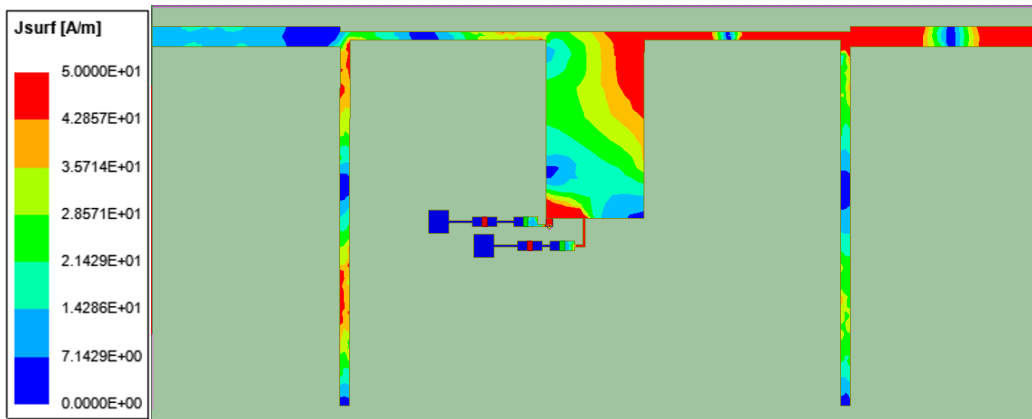
(d)
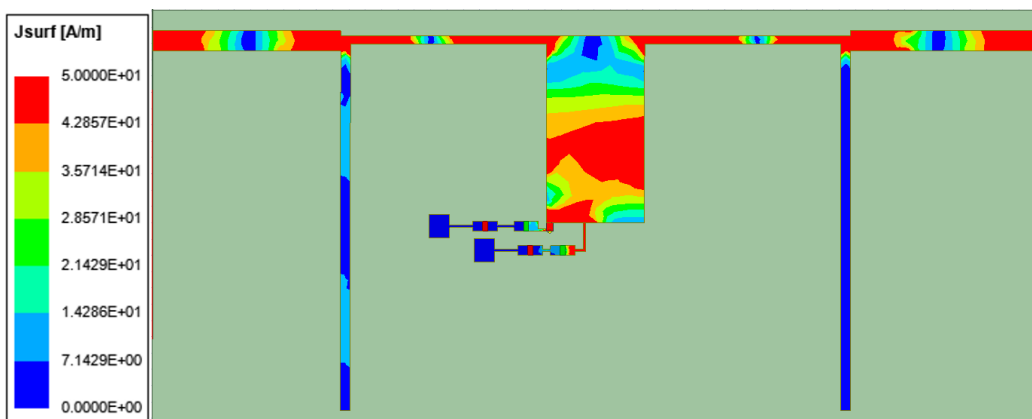
(e)
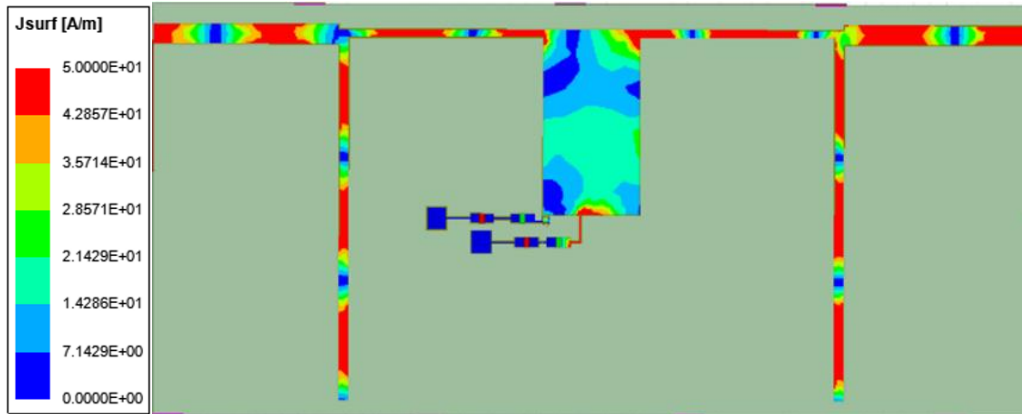
(f)

Figure 5. The simulated surface current distribution at frequencies (in GHz): (a) 0.01, (b) 1.23, (c) 2.14, (d) 4, (e) 5 and (f) 7.15.

Group delay is one of the filter's important characteristics. The simulated results for the first bandpass band are shown in Figure 6. The group delay changes tend to decrease by reducing the capacitance and its maximum value is 2.5 ns at 0 V, where the diode has the largest capacitance value of 4.15 pF.
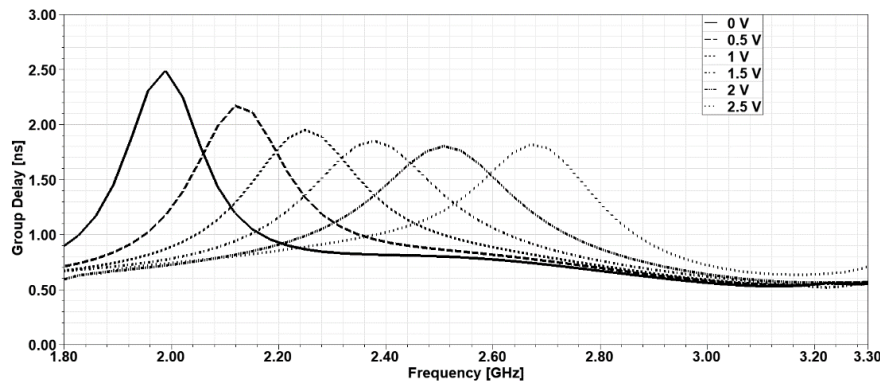
Figure 6. Group delay for the BPF 1st band.

Comparison between the proposed filter and different filters reported in published research that have been tuned using varactors regardless of the filter shape or the used elements to create the passing bands, is given in Table 3. The proposed filter has the largest number of passing bands compared to all other reported filters.

Table 3. Comparison between the proposed filter and filters presented in other published works.

Parameter Reference	Filter bands	No. of zeros	Tuning bands	FBW (%)	Side rejection (dB)	No. of varactors	Min. IL (dB)	Size (mm^3) (in λ^2)
[9]	BPF	3	2	11.29 – 14.77 8 – 9.38	> 40	6	3.86	19 × 33.35 × 0.79 (0.07λ × 0.16λ)
[11]	BPF	4	2	6.5 – 8.5 5.8 – 6.4	> 30	4	2.32	11.5 × 19 × 0.81 (0.04λ × 0.07λ)
[12]	BPF	2	1	4.1 – 7.1	> 50	2	5.0	25 × 35 × 1.524 (0.06λ × 0.08λ)
[13]	BPF	2	1	16.5 – 37.7	> 50	3	~ 4.5	30 × 54 × 0.508 (0.19λ × 0.34λ)
[14]	BPF	2	1	13.1 – 16.3	> 40	4	~ 4.0	40 × 40 × 0.8 (0.44λ × 0.44λ)
Proposed	**LPF & BPF**	6	**5**	2.74 – 8.4 1.55 – 9.93 5.54 – 15.38 6.64 – 8.12	> 30	**1**	**0.38**	52 × 108 × 0.8 (0.17λ × 0.35λ)

130

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

The number of zeros is introduced to show that these passbands are not harmonics of the first passband by noticing that all our passbands have an insertion loss lower than 3 dB with a lower sideband rejection greater than 30 dB over the entire considered frequency range. The proposed filter has a large substrate size compared to [9]-[11], because it is passing lower frequencies but is still small if measured in terms of the guided wavelength. The guided wavelength is calculated based on the lowest -3 dB intersection. It has the best IL values at the band center frequency compared to all the reported filters. Furthermore, the proposed filter has utilized only one varactor diode.

5. EXPERIMENTAL VERIFICATION

The designed filter is fabricated on an RT/Duroid 5880 substrate with a dielectric constant $\varepsilon_r = 2.2$ and height h = 0.8 mm as shown in Figure 7. The filter reflection and transmission coefficients have been measured by means of the vector network analyzer (Rohde & Schwartz ZNB8) [20] and are displayed in Figure 8 and Figure 9, respectively. The measured IL for BPF1 is from 0.76 – 2.44 dB and for BPF2 is from 0.78 – 2.94 dB. The parasitic resistor of the varactor accounts for this IL. The measured S_{21} and S_{11} compare favourably with the simulated results.
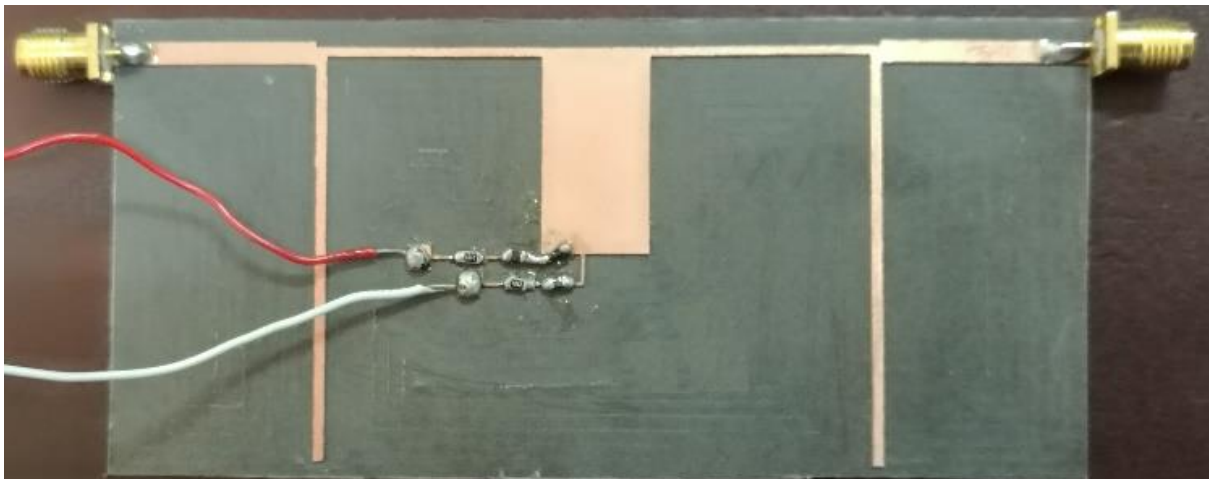
Figure 7. The fabricated filter with the DC biasing circuit.

Figure 8. Measured transmission coefficients for various varactor capacitances.
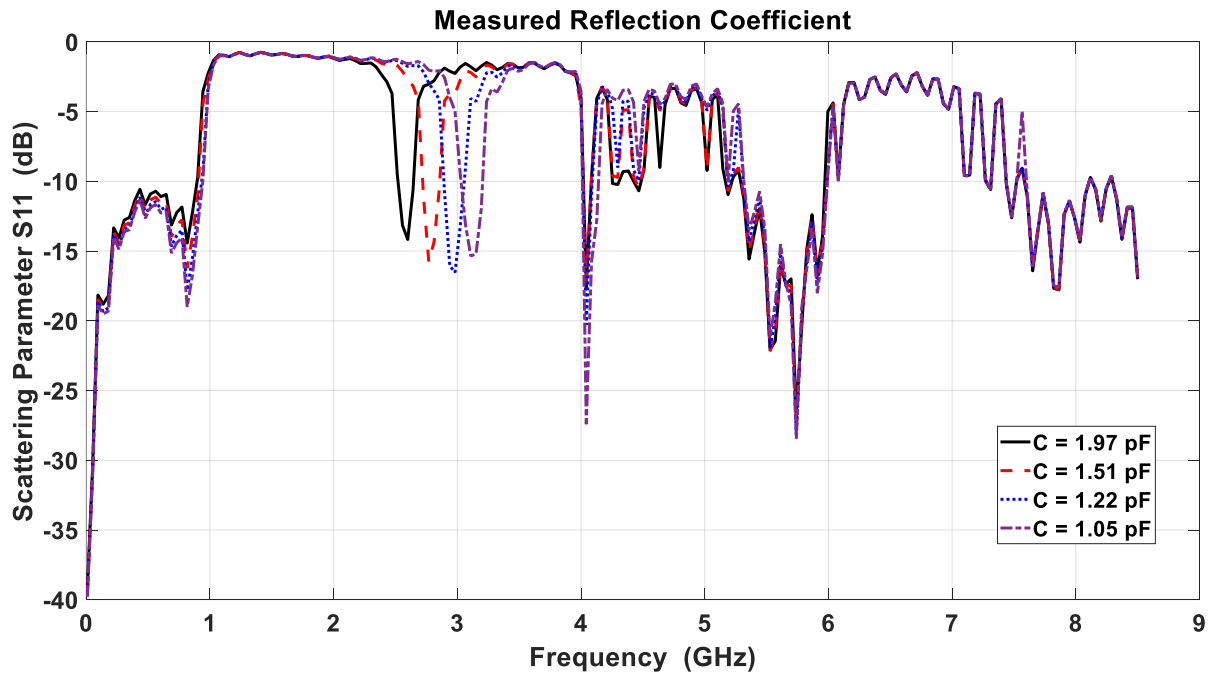
Figure 9. Measured reflection coefficients for various varactor capacitances.

The simulated and measured reflection coefficient results for the proposed filter under 2 V biasing voltage are shown in Figure 10 and they compare favourably. A comparison between the simulated and measured results is shown in Table 4. The differences between measured and simulated results are due to different factors which are not considered through the simulation process, such as the accuracy and precision of fabrication techniques used, the SMA connector welding, the non-homogeneous behaviour of the RT/Duroid substrate with frequency variations as well as the varactor behaviour with frequency. Varying the reverse biasing voltage will cause varactor capacitance to change and hence the center frequency and the bandwidth chane as well. The simulated and measured variation of the center frequency for the BPF1 with the reverse biasing is shown in Figure 11 (a).

The resonant frequency increases with increased reverse voltage. The bandwidth behaviour for the simulated and measured results with the varactor capacitance is shown in Figure 11 (b).
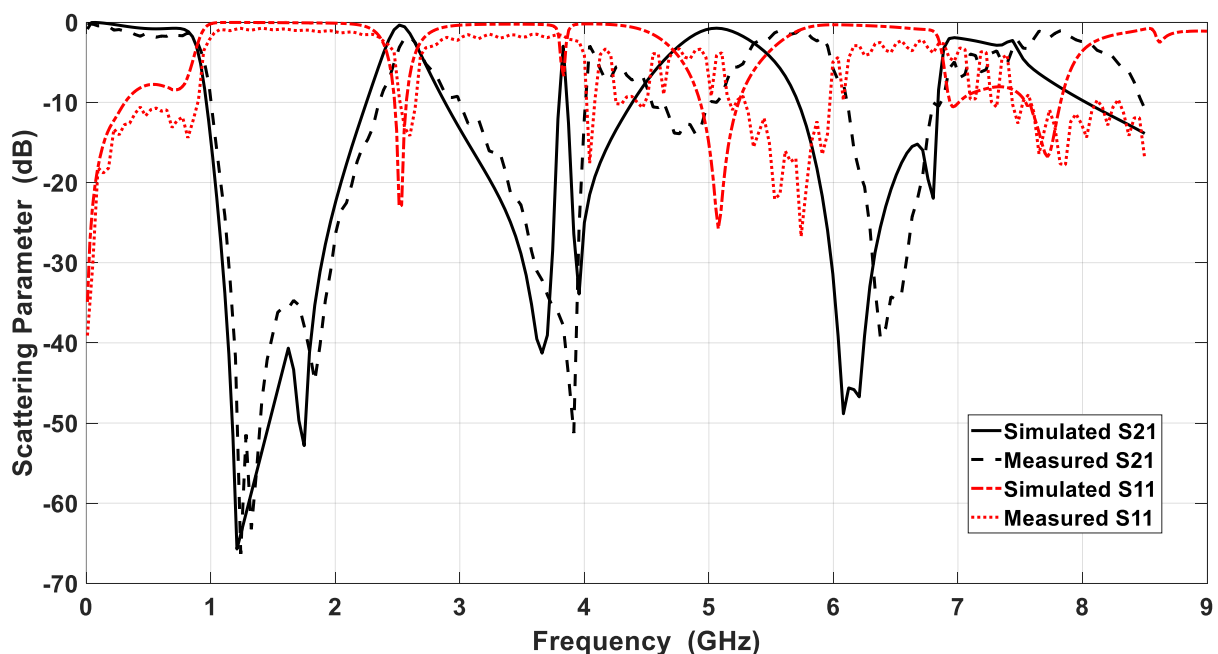
Figure 10. Simulated and measured S_{11} and S_{21} for C = 1.97 pF (V_R = 2 V).

(a) (b)
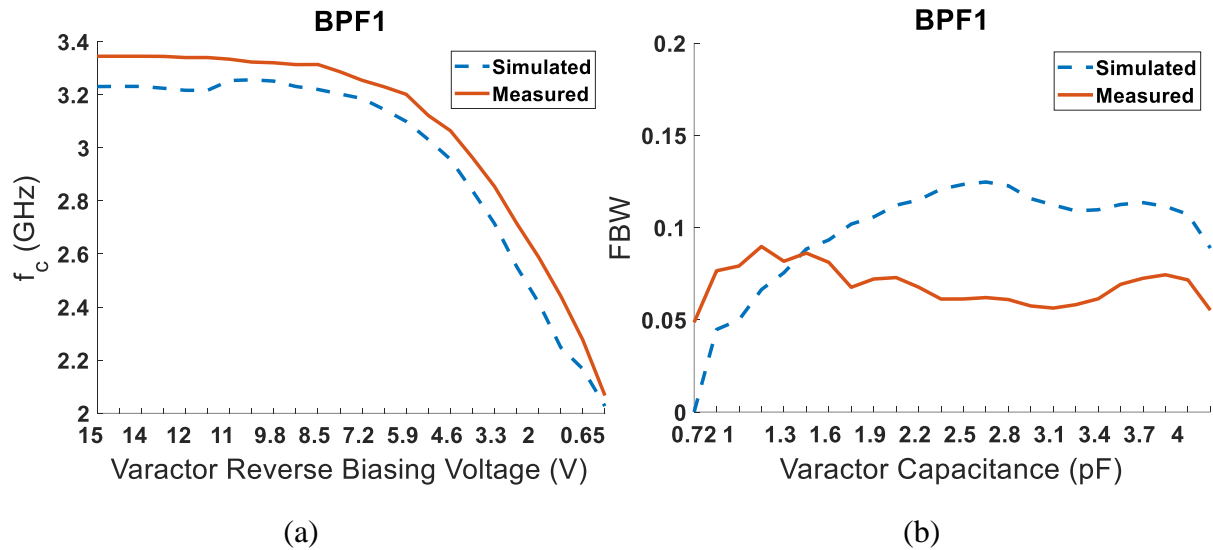
Figure 11. Simulated and measured results; (a) filter BPF1 center frequency *vs.* biasing voltage, (b) BPF1 fractional bandwidth *vs.* varactor capacitance.

Table 4. Comparison between simulated and measured results.

	Simulation			Measurement		
	Max IL dB	Tuning Range GHz	Zero level after the band	Max IL dB	Tuning Range GHz	Zero level after the band
LPF	---	0.71 – 0.94	-60	---	0.61 – 0.74	-53
BPF1	0.94	1.94 – 3.33	-39	2.44	2 – 3.4	-33
BPF2	1.24	3.83 – 4.23	-33	1.7	4 – 4.35	-22
BPF3	1.33	4.53 – 5.56	-36	1.83	5.35 – 5.95	-39
BPF4	1.8	6.83 – 7.48	---	2.3	7.4 – 8.15	---

6. CONCLUSION

A new planar enhanced tunable filter is designed for multi-band applications. The proposed filter consists of three stubs with two being open-circuited and a short-circuited stub in the middle that is connected to the ground through a shorting pin. A DC circuitry has been considered and implemented for the design of proper biasing of the varactor diodes. The design is investigated using HFSS. The simulation results show good impedance matching over the passing frequencies for RL ≥ 15 dB and lower insertion losses IL ≤ 3 dB. Results of measurements and simulation for the transmission and reflection coefficients compare favourably. Current distribution at different selected frequencies has been viewed to clarify the filter behaviour. The filter group delay is lower than 3 ns. This kind of filters could be useful in future reconfigurable RF front-end systems for radars and wireless communication systems.

ACKNOWLEDGEMENTS

REFERENCES

[1] P. W. Wong and I. Hunter, "Electronically Tunable Filters," IEEE Microwave Magazine, vol. 10, no. 6, pp. 46–54, 2009.

[2] J. Uher and W. J. R. Hoefer, "Tunable Microwave and Millimeter-wave Band-pass Filters," IEEE Transactions on Microwave Theory and Techniques, vol. 39, no. 4, pp. 643–653, 1991.

[3] I. C. Reines et al., "A Low Loss RF MEMS Ku-band Integrated Switched Filter Bank," IEEE Microw.

Wirel. Components Lett., vol. 15, no. 2, pp. 74–76, 2005.

[4]   C. A. Hall, R. C. Luetzelschwab, R. D. Streeter and J. H. Vanpatten, "A 25 Watt RF MEM-tuned VHF Bandpass Filter," Proc. of IEEE MTT-S International Microwave Symposium Digest, pp. 503–506, 2003.

[5]   J. H. Park, H. T. Kim, Y. Kwon and Y. K. Kim, "Tunable Millimeter-wave Filters Using a Coplanar Waveguide and Micromachined Variable Capacitors," Journal of Micromechanics and Microengineering, vol. 11, no. 6, pp. 706–712, 2001.

[6]   A. Pothier et al., "Low-loss 2-Bit Tunable Bandpass Filters," IEEE Transactions on Microwave Theory and Techniques, vol. 53, no. 1, pp. 354–360, 2005.

[7]   E. Fourn et al., "MEMS Switchable Interdigital Coplanar Filter," IEEE Transactions on Microwave Theory and Techniques, vol. 51, no. 1, pp. 320–324, 2003.

[8]   C. S. Tsai, "Wideband Tunable Microwave Devices Using Ferromagnetic Film-gallium Arsenide Material Structures," Jou. Magn. Magn. Mater., vol. 209, no. 1–3, pp. 10–14, 2000.

[9]   X. Zhang, C. Chen, M. Li, L. Zhou and B. Liu, "A Compact Tunable Dual-band Bandpass Filter Using Varactor-loaded Step-impedance Resonators," PIERS Proceedings, vol. 1, pp. 2642–2645, Prague, Czech Republic, 2015.

[10]   L. Smadi, H. Sharif and Y. S. Faouri, "Dual-Band Tunable Microwave Bandpass Filter Using Stepped Impedance Technique," Proc. of IEEE Jordan Int. Jt. Conf. Electr. Eng. Inf. Technol., pp. 827–830, Apr. 2019.

[11]   X. Y. Zhang, L. Gao, Y. Cao, X.-L. Zhao and Y. Ding, "Independently-tuned Dual-band Filter Using Varactor-loaded Resonators," Prog. Electromagn. Res. C, vol. 42, no. May, pp. 55–66, 2013.

[12]   A. Golaszewski, M. Zukocinski and A. Abramowicz, "Design of Varactor Tuned Bandpass Filter," Proc. of the 14th Conf. Microw. Tech. Com., pp. 86–89, 2015.

[13]   Y. Guan and Q. Guo, "A Varactor-tuned Bandpass Filter Based on Lowpass Filter and Shorted Stubs," Microw. Opt. Technol. Lett., vol. 56, no. 4, pp. 879–883, 2014.

[14]   B. Kim and S. Yun, "Varactor-Tuned Combline Bandpass Filter," IEEE Trans. Microw. Theory Tech., vol. 52, no. 4, pp. 1279–1283, 2004.

[15]   Y. Shang, W. Feng and W. Che, "Wideband Reconfigurable Bandpass Filter Using Coupled Lines Loaded with Varactor Loaded Stubs," Int. J. RF Microw. Comput. Eng., vol. 28, no. 2, 2018.

[16]   S. Arain, P. Vryonides, M. A. B. Abbasi, A. Quddious, M. A. Antoniades and S. Nikolaou, "Reconfigurable Bandwidth Bandpass Filter with Enhanced Out-of-Band Rejection Using π-Section-loaded Ring Resonator," IEEE Microw. Wirel. Components Lett., vol. 28, no. 1, pp. 28–30, 2018.

[17]   H. Sharif, L. Smadi and Y. S. Faouri, "Stub Resonator Tunable Bandpass and Lowpass Filters Using Shunt Stub Resonators," Proc. of IEEE Jordan Int. Jt. Conf. Electr. Eng. Inf. Technol., pp. 442–445, Apr. 2019.

[18]   D. M. Pozar, Microwave Engineering, 4th Edition, New York: Wiley, 2012.

[19]   Skyworks Solutions, "SMV123x Series : Hyperabrupt Junction Tuning Varactors," [Online], Available at: http://www.skyworksinc.com/uploads/documents/200058Q.pdf, 2012.

[20]   W. D. Reeve, "Filter Measurements with a Vector Network Analyzer," pp. 1–16, [Online], Available at: http://www.reeve.com/Documents/Articles%20Papers/Reeve_FilterMeasVNA.pdf, 2016.

ملخص البحث:

تــم تصـميم مرشِّــح قابــل للضــبط الكترونيــاً لتمريــر عــددٍ   مــن النطاقــات الترددية والــذي يســتخدم ثنائيــاً متغيــر المواســعة وتنفيــذه بنــاءً علــى تقنيــة رنــين الصَّــدم، وذلــك مــن أجــل تمرير عدة تطبيقاتٍ مفضَّلة عبر نطاقات تشغيل ترددية متعددة.

فــي هــذه الورقــة، تــم تصــميم المرشِّــح بحيــث كــان طرفــا مدخلــه ومخرجــه مــزوَّدين بخــط تغذيــة شــريطي صــغير مقاومتــه (50) أوم. أمــا النطاقــات الترددية التــي صُــمِّم المرشِّــح لتمريرهــا فهــي: نطــاق للتــرددات المنخفضــة بتــردد قطــع قابــل للضــبط بحيــث يصــل الــى (0.94) جيجــاهيرتز، بالإضــافة الــى عــدد مــن النطاقــات الترددية الأخــرى التــي تتــراوح ترددداتهــا بــين 1.94-3.3 جيجــاهيرتز للنطــاق الأول، وبــين 3.83-4.23 جيجــاهيرتز للنطــاق الثــاني، وبــين 4.53 -5.56 جيجــاهيرتز للنطــاق الثالــث، وبــين 6.83-7.48 جيجــاهيرتز للنطــاق الرابــع. أمــا فقــد الإدخــال للمرشِّــح عنــد تلــك التــرددات، فهــو يقــلّ عــن (3) ديســيبل. والجــدير بالــذكر أن المرشِّــح المصــمم فــي هــذه الدراســة هــو مــن النــوع ذي الحــدَّين، ويتضــمن (3) عناصــر منفّــذة فــي شــكل صَــدّامات تــوازٍ بحيــث تكــون الصــدّامة الوسطى في حالة قِصَر.

وقــد أجريــت دراســة لتحديــد الموضــع المثــالي للثنــائي متغيــر الســعة، واســتخدم مصــدر خــارجي لتــوفير الانحيــاز العكســي المطلــوب للثنــائي متغيــر الســعة وأخــذ تــأثيره بعــين الاعتبــار. وتــم اســتخدام محــاكي بِنيــة التــرددات العاليــة (HFSS) لدراســة المرشِّــح المصــمم فــي هــذه الدراســة. وكانــت النتــائج المقاســة المتعلقــة بمعــاملات التشــتت (معامــل الانعكــاس ومعامــل الارســال) متفقــة بصــورة جيــدة مــع القيــم التــي تــم الحصــول عليهــا عــن طريق المحاكاة.

135

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

A Parallel Pipelined Packet Switch Architecture for Mesh-connected Multiprocessors with Independently Routed Flits

Jamil Al-Azzeh[1], Mohammed Agmal[2] and Igor Zotov[2]

ABSTRACT

In this paper, a packet switch architecture for mesh-connected multiprocessors based on the use of a set of input FIFO buffers and an output register matrix controlled by a novel distributed timing-based scheduling scheme is proposed. Simple static routing is assumed, with each packet split into a set of independently routed w-bit-wide flits. The device achieves at least 78% throughput for uniformly distributed traffic and an asymptotic higher bound of 100%. In contrast to the state-of-the-art VOQ-based switch architectures, the proposed switch is shown to reach its maximum throughput with no internal speedup required and has an order of magnitude lower hardware complexity. Compared to existing buffered crossbar non-VOQ switches with typical flit scheduling mechanisms, the proposed device demonstrates slightly higher throughput and substantially shorter delays in some practically important cases.

KEYWORDS

Multiprocessor, Mesh topology, Packet switching, Input-queued switch, FIFO-buffer, Flit, Pipelining, Throughput.

1. INTRODUCTION

Switching hardware is known to play a crucial role in the operation of a wide class of modern computer systems [1]. Mesh-connected multiprocessors are an example of systems whose performance is profoundly affected by the underlying built-in switching apparatus [2]. Inter-processor data exchange speed and remote memory access latency significantly depend on the throughput and performance of the switches distributed across the multiprocessor mesh [37]-[39].

The packet switching paradigm is the most widely used in contemporary multiprocessor designs, such as chip multiprocessors (e.g., see [3]-[5] and the references therein). Furthermore, hybrid (packet/circuit) switching approach has evolved as well (e.g., see [6]-[9]). Packet switches utilized in modern multiprocessors are in several respects similar to the asynchronous transfer mode switches employed in computer networks and supercomputers [1]. Various switch architectures mainly differ in the arrangement of internal packet buffers (queues). In input-queued switches, packets are first loaded into the corresponding input buffers and then switched to the required outputs *via* a crossbar. Such switches do not impose hard requirements for the internal speed of the switching hardware (crossbar) and, as a rule, have relatively low hardware complexity, which allows them to be utilized with a large number of inputs/outputs. However, it has been shown [10] that input-queued switches with simple FIFO buffers are limited in their throughput at approximately 0.64 in practically significant scenarios (or at $2-\sqrt{2} \approx 0.586$ asymptotically) due to the occurrence of blocked packets in the head cells of the buffers (known as HOL blocking). The highest possible throughput (up to 100%) is achieved in output-queued switches. In such devices, packets are immediately switched to the required outputs and then stored in the output buffers before being issued, which eliminates HOL blocking. However, output-buffered switches impose strict requirements on the internal speed of the switching apparatus, which must be several times higher than the external speed at which packets travel between processors. In this regard, their use is feasible with a relatively small number of inputs/outputs *n* and/or if the external switch speed is deliberately decreased.

To overcome the problems inherent to input- and output-queued switches with ordinary FIFO buffers,

---

1. J. Al-Azzeh is with Dep. of Computer Eng., Al-Balqa Applied University, Al-Salt, Jordan. Email: `azzehjamil18@gamil.com`
2. M. Agmal and I. Zotov are with Southwest State University, Russia. Emails: `MohammedAgmaAbdo@gamil.com` and `zoto-vigor@yandex.ru`

several solution methods have been proposed in the recent past (e.g., see [11]-[16] and the references therein). Most of these methods are based on the virtual output queue (VOQ) paradigm first proposed in [17]. In a VOQ switch with n input and n output trunks, n separate simple FIFO buffers are combined in parallel at each input port, each of which corresponds to a given output. Packets arriving at a particular input are immediately routed to an input queue corresponding to the required output and are then switched to the output line *via* a crossbar circuit; therefore, packets of the same input queue never require to be transferred to different outputs and thus HOL blocking never occurs.

In order to maximally improve the VOQ switch throughput, a central scheduler is required for contention resolution when two or more packets are destined to the same output at the same time slot (e.g., see [18]-[19]). Moreover, to achieve 100% throughput (or non-blocking performance), a stable scheduling policy is needed for any admissible traffic pattern. It is worth noting that adding extra crosspoint buffers to the internal crossbar enhances the switch throughput as well [20]. Severe VOQ switch scheduling algorithms, such as maximum size/weight matching [21], have been introduced and studied. Although these algorithms are shown to be stable, they are too complicated for efficient hardware implementation in multiprocessors. Consequently, suboptimal solution algorithms are usually adopted. The underlying idea is to find an input-to-output matching of the maximal size (known as MSM) with no input or output left unnecessarily idle. Finding an MSM is more efficient, because it does not require backtracking. Iterative scheduling algorithms for finding an MSM have been widely adopted (e.g., see [22]-[24] and the references therein). An iterative scheduling algorithm needs to execute up to n iterations to guarantee maximal size match. However, as far as we know, all existing stable iterative algorithms require a speedup of 2, approximately; i.e., the internal speed of the switch must be twice the external speed at which packets are transferred between processors. A lower speedup is always desirable, because it reduces the implementation cost and increases the external speed of the switch at the same time.

Furthermore, a load-balanced two-stage-based architecture has been studied in high-speed switch design and employed to maximize the switch throughput (e.g., see [25]-[26]). Load-balanced two-stage switches are excellent techniques of getting rid of the central scheduler and reducing hardware-level complexity. Nevertheless, their main drawback is that packets may be eventually mis-sequenced. Further, load-balanced switches suffer from high delay performance under low to medium load. Thus, the main challenge of contemporary switch technology is still how to improve the efficiency of the central scheduler significantly.

Although there are no major barriers in the use of wormhole-routed VOQ-based packet switches in mesh-connected multiprocessors, specific issues and limitations arise that make the usage less efficient compared to multi-computers and computer networks. Because multiprocessor nodes exchange short packets split into a small number of w-bit-wide flits transferred *via* trunks each in one clock cycle, the VOQ scheduling time might become a significant portion of the packet transfer duration. The other issue is that these packet switches require the internal speedup of about 2, which leads to significantly higher implementation cost and makes it impossible to set state-of-the-art external speed values. In this regard, it is worth mentioning the Epiphany IV VLSI-multiprocessor architecture as an example, which has 136-bit-wide packets with 64 bits of data, 64 bits of address and 8 bits of control [5].

The independently routed flit (IRF) paradigm is one of the alternatives to the traditional wormhole-routed VOQ-based networking for mesh-connected multiprocessors. With the IRF approach, a packet is divided into a set of flits of the same width w, each of which carries both data and address fields in addition to some control/identification bits (e.g., see [27]-[28]). In contrast to wormhole-routing, in the IRF approach, flits are not grouped to be processed and transferred as an atomic entity; instead, they are routed independently based on a simple static strategy. After arriving at a given destination, flits are lined up (reassembled) in a packet. If flits arrive in a wrong order, they can be fetched by their identifiers to restore the initial ordering, which is a deadlock and livelock free process. If one or more flits are missing, a timeout mechanism can help solving this reliability-related problem with no permanent blocking taking place. The main advantage of the IRF networking is that it provides more simple hardware solutions than wormhole virtual cut-through routing, because no specific wormhole-aware algorithms and apparatus are required to support virtual channels and to guarantee deadlock and livelock freedom.

In this paper, we consider mesh-connected multiprocessors similar to the Epiphany IV [5] or the Tile-Gx series [4]. Moreover, we employ the IRF-networking-based switch architecture with w-bit-wide flits

137

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 05, No. 02, August 2019.

and simple static routing [28]. Our main contribution is the proposition of a parallel pipelined switch architecture (which we further refer to as the PPIRF switch) based on the use of non-VOQ input FIFO buffers and an output register matrix (buffered crossbar) controlled by a built-in distributed flit scheduler implementing a novel row-wise oldest-flit-first discipline formalized based on the construction and manipulation of flit consistency graphs. The proposed architecture achieves the throughput of at least 78% for all practically significant scenarios and 100% asymptotic switch throughput (assuming Bernoulli uniform traffic). The major effect gained is that neither virtual queuing at input ports, nor internal speedup is required to achieve up to 100% asymptotic throughput, which leads to a quadratic asymptotic hardware complexity—an order of magnitude less than that of VOQ-based switches. Compared to similar buffered-crossbar-based switches, the PPIRF switch demonstrates slightly higher throughput and significantly shorter delays in some cases (e.g., for heavier flit traffic patterns), resulting from the novel scheduling policy employed. In what follows, we formally state the structural organization and operation of the proposed parallel pipelined switch; further, we present and briefly discuss some simulation results; finally, we make some comparison and present concluding remarks and future directions as well.

2. THE PARALLEL PIPELINED IRF SWITCH ARCHITECTURE

Figure 1 shows a formalized structural model of the parallel pipelined IRF switch. The switch has several external connections (trunks)—a set of inputs $I_1, I_2, \ldots, I_n$ and a set of outputs $O_1, O_2, \ldots, O_n$—and is composed of input FIFO buffers (queues) $Q_1, Q_2, \ldots, Q_n$ of length (size) L, a register matrix $B = \|B_{ij}\|$, $i, j = \overline{1, n}$, demultiplexers $R_1, R_2, \ldots, R_n$, multiplexers $K_1, K_2, \ldots, K_n$, $M_1, M_2, \ldots, M_n$ and gates $G_1, G_2, \ldots, G_n$. In Figure 1, the solid lines represent flit transfer paths, while the dashed lines indicate the control signal paths.

Input I_1 and output O_1 have a predefined function to connect the switch to the corresponding processor core (current tile). The rest of the inputs and outputs link the current node to the neighboring nodes in the multiprocessor mesh. This means that for a two-dimensional (2D) mesh multiprocessor, a 5×5 switch is required ($n = 5$) at each node; in turn, a 3D mesh system would require the use of a 7×7 switch ($n = 7$).

The functions of the blocks in Figure 1 are as follows. The input FIFO buffers $Q_1, Q_2, \ldots, Q_n$ are used to temporarily store flit streams arriving at the corresponding input trunks before they are transferred to the register matrix B. It has to be mentioned that the size (L) of the buffers is specified at the switch implementation stage and is assumed to be arbitrary. Each input has the only associated FIFO buffer structured around a set of static registers, each of which is capable of holding a single w-bit-wide flit. The set of demultiplexers $R_1, R_2, \ldots, R_n$ implement a predetermined flit routing algorithm μ. The function of the register matrix B is to automatically distribute incoming flit streams between its rows following the output directions obtained from the routing algorithm. The multiplexers $M_1, M_2, \ldots, M_n$ implement a given flit scheduling scheme φ (the oldest-flits-go-out-first policy, in our case), which is presented in detail below. Furthermore, the multiplexers $K_1, K_2, \ldots, K_n$ together with the gates $G_1, G_2, \ldots, G_n$ are needed to temporarily block flits from being transferred to the register matrix from the corresponding queues if their respective target registers still contain unissued flits (HOL blocking).

3. THE OPERATION OF THE PIRF SWITCH

The operation of the PPRF switch is generally organized in four cyclically repeating steps, which are as follows:

1. Determine the output directions for all the flits located in the head cells of the input queues (further referred to as the head flits) based on the routing algorithm μ.
2. Transfer the head flits from the respective input queues to the register matrix *B*, unless the corresponding target registers in the register matrix contain any other flits; then shift the queues in the case of a successful transfer.
3. Analyze the current flits-to-registers mapping and select a subset of flits, which must be issued from the register matrix based on the scheduling scheme φ taking into account the time elapsed from the

138

moments of their arrival.

4. Issue the selected subset of flits to the respective outputs and reset the corresponding registers of the register matrix *B*.
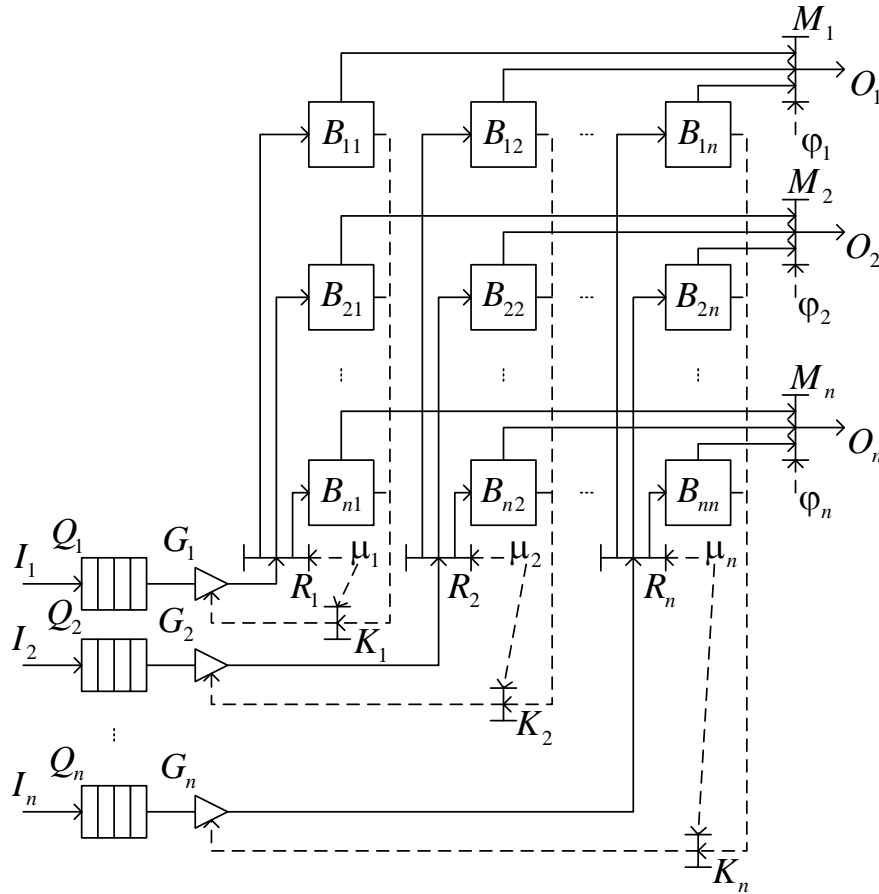

Figure 1. A formalized structural model of the parallel pipelined IRF switch.

Figure 2 shows a detailed parallel flow-chart representing the operation of the PPIRF switch. The flow-chart uses some formal constructs and symbols, whose meanings are explained below.

In Figure 2, F_k stands for the actual set of flits distributed between the registers of the register matrix *B,* where *k* denotes the current cycle on the switch operation timeline. The flit consistency graph Γ_k with a set of vertices F_k and a set of edges $\alpha_k \subseteq F_k \times F_k$ is introduced to indicate whether a pair of flits can be issued from the register matrix in parallel. The following rule formally defines the flit consistency relation α_k:

$$\left(f_q, f_r\right) \in \alpha_k \Leftrightarrow f_q \in \bigcup_{j=1}^{n} S_k\left(B_{i'j}\right), f_r \in \bigcup_{j=1}^{n} S_k\left(B_{i''j}\right): i' \neq i'', \tag{1}$$

where S_k is an indicator function, such that $S_k\left(B_{ij}\right) = f_q$ if the register B_{ij} contains the flit f_q before the kth cycle begins; and $S_k\left(B_{ij}\right) = \varnothing$ if the register B_{ij} is empty. Each vertex of the graph Γ_k is given a non-negative weight τ_q, which reflects the time elapsed since the flit f_q had arrived at the respective input of the switch (measured in cycles). A clique $\Gamma'_k = \langle F'_k, \alpha'_k \rangle$, $F'_k \subseteq F_k$, $\alpha'_k \subseteq \alpha_k$ in the graph Γ_k is selected, such that the total weight of its vertices becomes the maximum across the set of candidate cliques:

$$\sum_{f_q \in F'_k} \tau_q = \max. \tag{2}$$

If the graph Γ_k contains a set of cliques, for each of which condition (2) holds, then any clique Γ'_k of this set is picked out assuming uniformly distributed random selection process. It is evident from the

above formal statements that the vertices F_k' of the clique Γ_k' are consistent based on (1) and the cardinality of F_k' is maximal by inclusion.

In Figure 2, we also employed a new indicator function s, such that $s(Q_l) = 1$ if the queue Q_l currently contains at least one flit and $s(Q_l) = 0$ otherwise. The symbol $\leftarrow$ denotes register/trunk read/write operations. For example, $f_i \leftarrow \text{head}(Q_i)$ means the extraction of the head flit f_i from the queue Q_i and $O_i \leftarrow f_{j_i}$ implies that flit f_{j_i} should be issued to output O_i. Additionally, $\mu(f_i)$ stands for the direction (i.e., the output trunk) that flit f_i is supposed to be routed based on the algorithm μ.

The operation of the PPIRF switch according to the algorithm of Figure 2 is organized in a series of cycles (loops). Each cycle (k^{th} cycle) consists of vertices 3 to 14 and is performed within a single time slot.

First, condition 3 is checked. If at least one input queue contains flits, the condition holds and the loop begins. Otherwise, the algorithm terminates and the switch keeps staying idle.

Each cycle starts with a parallel section composed of n threads. The ith thread ($i = \overline{1, n}$) consists of vertices 4.i-8.i and describes how the flits are fetched and transferred from queue Q_i to the ith column of the register matrix. First, a flit (referred to as f_i) is read from the head register of queue Q_i (vertex 4.i). Second, the output trunk (and the corresponding target register of the register matrix) is determined to which f_i has to be relayed according to the supported routing algorithm (vertex 5.i). Third, the HOL blocking condition (vertex 6.i) is checked. If the target register still contains another flit, the ith thread terminates and the state of queue Q_i remains unchanged. Otherwise, flit f_i is extracted from the head cell of queue Q_i and immediately uploaded to the target register. Then, Q_i is shifted by 1 position.

As soon as all the threads have terminated, the loop proceeds with vertices 9 and 10. According to vertex 9, graph Γ_k is formed based on the current flit-to-register distribution. In fact, it is a dummy operation, because the graph is automatically formed immediately after all the arrived flits have been mapped onto the register matrix. Thus, it takes no extra time. Then, according to vertex 10 of the algorithm, a clique of the graph is selected based on formula (2). In the selection process, each row of the register matrix is examined separately and in parallel to its peers. In each row, a flit is picked out whose vertex in graph Γ_k has the maximum weight amongst the flits mapped onto the same row. This process has $O(n)$ runtime complexity if implemented sequentially. In the case of parallel calculation of the maximum, the runtime complexity is $O(1)$ at the cost of parallel hardware. If a pyramidal maximum computation circuit is employed, the process will take $O(\log n)$ time units. Because the rows of the register matrix are examined in parallel, the entire process of picking out a clique requires $O(1)$ time in the best case (parallel scheme) and $O(n)$ in the worst case (sequential scheme). Immediately after a clique has been picked out, another parallel section begins (vertices 11.j-12.j). In accordance with the jth thread of this section, the selected flit of the jth row (the one whose vertex belongs to the clique) is transferred to the jth output trunk and the corresponding register is immediately reset (vertex 12.j).

Hence, the final part of the loop commences, which consists of vertices 13 and 14. In line with vertex 13, the current graph Γ_k is reconstructed by eliminating all the vertices that correspond to the relayed flits and by incrementing the weights of all the remaining vertices corresponding to the flits that could not be selected and transferred within the kth loop. And finally, k is incremented (vertex 14) and the next loop starts.

Figure 3 gives an example of how flit consistency graphs are constructed and treated in the operation of a PPIRF switch according to the algorithm of Figure 2 (vertices 9 and 10). In Figure 3, a 5×5 switch is considered and the state of the register matrix for 9 consecutive cycles (marked with letters a, b, …, i) of the algorithm is presented. Hereinafter in the example, the squares and the circles denote the registers of the register matrix and the flits (vertices of graphs Γ_k) being processed, respectively;

the designation f_i^k stands for a flit extracted from queue Q_i in cycle k; the weights τ_q of the flits are placed inside the corresponding circles.
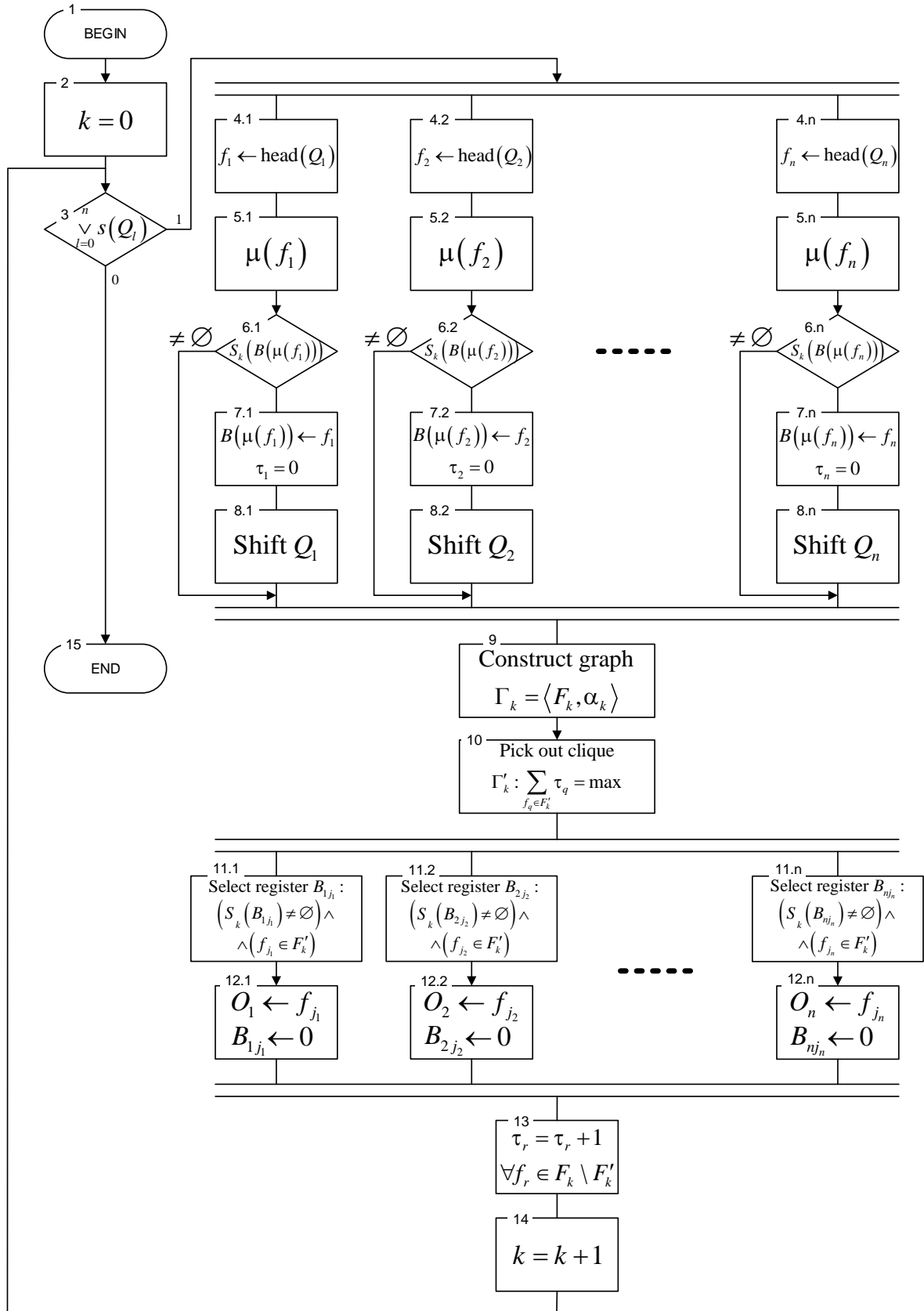

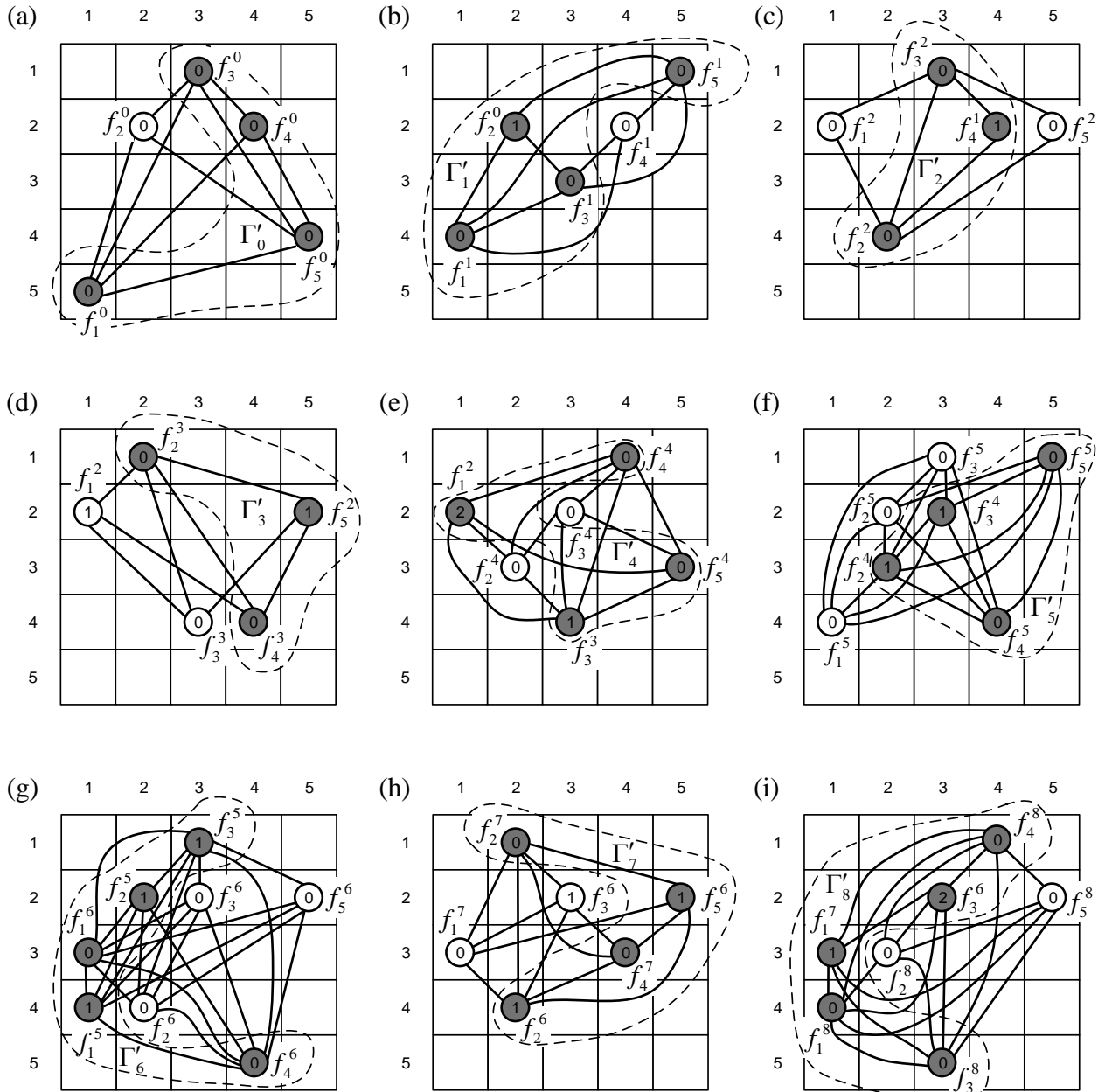Figure 2. A parallel flow-chart representing the operation of the PPIRF switch.

Figure 3. Example of flit consistency graph construction and manipulation in the operation of a 5×5 PPIRF switch according to the algorithm of Figure 2.

Figure 3(a)-(i) demonstrates how the cliques $\{\Gamma'_k\}$ (encircled in dashed curves) of the corresponding flit consistency graphs $\{\Gamma_k\}$ are picked out according to (2) and how the graphs are reconstructed as the process evolves step by step. Note that new flits are added to the register matrix at random as if they were directed using the routing algorithm μ.

4. SIMULATION STUDIES AND RESULTS

We conducted a series of comprehensive simulation studies to investigate the throughput, delay and some other characteristics of the proposed PPIRF switch and compare our solution with similar buffered crossbar switches supporting typical packet scheduling policies (uniform and round robin) [29]-[30]. Based on the extended Q-chart modeling language—in a similar way as utilized in [31]—we constructed a queuing model that represents the behavior of the PPIRF switch and the buffered peers under consideration. The choice of the language we used is determined by the presence of simulation entities such as group controls, which greatly simplify the implementation of various switching functions; for example, multiplexing and demultiplexing, which allows creating quite compact models for complex devices and

systems. To perform the simulation experiments, we employed a dedicated Q-chart-based simulation testbed (Visual QChart Simulator) developed by Zotov et al. [32]. The Visual QChart Simulator integrated environment enables the user to input Q-chart models as graphics, automatically check and compile them to C++ code. Further, it generates an executable using a C++ command line compiler/linker chain to simulate the behavior of a given unit/system with preset to simulation parameters such as simulation duration, number of flits issued and flit traffic characteristics, to mention but a few.

All the switches examined were assumed to be queuing networks, in which flits arriving at the inputs were considered as random service request streams. Typically, Bernoulli uniform, Bernoulli hot spot and bursty uniform traffic patterns are assumed when studying the throughput and performance of the switch architectures [10], [19], [24] and [29]. However, bursty traffic is specific to the ATM-based networks and uncommon for the multiprocessors considered in the paper; therefore, we studied the uniform and hot spot traffic patterns only. Short-term traffic asymmetry was admitted. In our experiments, we set up all simulation conditions as done in [10]. Considering Bernoulli uniform traffic, the probability that a flit arrives at the ith input of the switch $\left(i = \overline{1,n}\right)$ in the next cycle was assumed to be p; thus, the probability of no packet arrival is $q = 1 - p$. The supported routing algorithm μ was considered a *priori* unknown and therefore we supposed uniformly distributed selection of the register matrix target rows to transfer flits from the input buffers.

In the simulation experiments, the number of inputs/outputs n was selected from the range of 5 to 25 with one step increment, which corresponds to practically significant scenarios for modern multiprocessors such as mesh [4]-[5], torus [33], cube [34], twisted torus [35] and crossed cube [36]. We also studied next-generation switches separately, with $n \gg 25$ (up to 1000) and set the Bernoulli distribution parameter p equal to 1 for all n inputs to evaluate the throughput of the switch. The duration of the simulation study was assumed to be 10,000 switch flit relay cycles. Besides, the required number of repetitions with fixed parameters was determined by the Student's criterion for the significance level of $\alpha = 0.02$.

Figure 4 shows the 5×5 switch Q-chart utilized in our experimental studies (note that the same Q-chart structure is used to model both the PPIRF switch and existing buffered crossbar peers [29]-[30], but different control logic is employed reflecting the corresponding scheduling policies). Five types of entities (elements) are utilized in the Q-chart: flit generators (denoted as Gx), flit processors (Dx), queues (Qx), group controllers (RCx and MCx) and gates (Kx). Table 1 presents a list of the functions of the simulation entities. In the chart of Figure 4, the solid lines denote the information links which fix flit transmission paths and the dashed lines show control signals (links) that specify the conditions affecting the state of the controlled simulation elements that are letting flits pass through or blocking them temporarily. Arrows specify in which directions flits or control signals are transmitted. Furthermore, both information and control links are unidirectional.

The gates are a cornerstone of the Q-chart logic. For the chart shown in Figure 4, the following gate enable/disable rules apply:

- one gate is enabled (open) only among gates K1i1, K2i1, K3i1, K4i1 and K5i1 ($i = \overline{1,5}$), with the probability of 0.2 (or the probability of $\frac{1}{n}$ for arbitrary n);

- gate Kji2 ($i, j = \overline{1,5}$) is enabled (open) if queue Qji2 is empty;

- gate Kji3 ($i, j = \overline{1,5}$) is enabled (open) subject to Qji2 containing a flit waiting to be issued (depending on the time the flit has spent in the switch);

- gate K_LOAD_i ($i = \overline{1,5}$) is enabled (open) if all queues Q1i1, Q2i1, Q3i1, Q4i1 and Q5i1 are empty;

- gate K_LOST_i ($i = \overline{1,5}$) is enabled (open) only if queue Q_i has a limited capacity and is full of flits (this is useful to identify the conditions when the switch starts losing incoming flits).
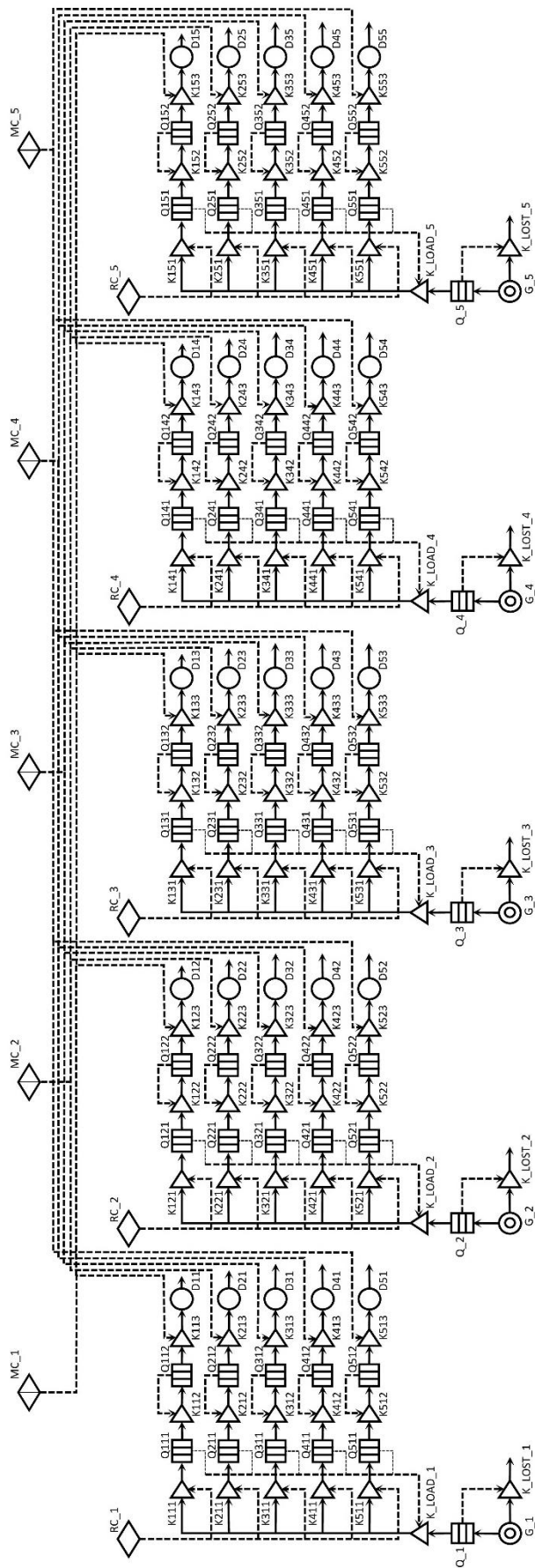
Figure 4. Q-chart representing a switch with five input and five output trunks.

Table 1. The functions of the simulation entities of the Q-chart given in Figure 4.

No.	Element ID	Element function
		Flit generators
1	G_i ($i = \overline{1,5}$)	They simulate a flit stream that arrives at input I_i (input traffic).
		Flit processors
2	Dji ($i, j = \overline{1,5}$)	They simulate a flit stream initially arrived at input I_i and now leaving the jth row of the register matrix *via* output O_j (output traffic).
		Queues
3	Q_i ($i = \overline{1,5}$)	They represent input queues Q_i.
4	Qji1 ($i, j = \overline{1,5}$)	They simulate the transferring of a flit from the queue Q_i to the ji^{th} register of the register matrix.
5	Qji2 ($i, j = \overline{1,5}$)	They simulate the storing of a flit in the ji^{th} register of the register matrix for flits initially uploaded from the queue Q_i
		Group controllers
6	RC_i ($i = \overline{1,5}$)	They select a row of the register matrix based on the routing algorithm μ (with equal probabilities in this study).
7	MC_j ($j = \overline{1,5}$)	They select a register in the jth row of the register matrix to issue the flit in accordance with the algorithm φ (oldest flits are issued first in this study).
		Gates
8	Kji1 ($i, j = \overline{1,5}$)	They enable/disable the selection of the queue Q_i before uploading its head flit to the ji^{th} register of the register matrix.
9	Kji2 ($i, j = \overline{1,5}$)	They enable/disable the upload of the head flit of the queue Q_i to the ji^{th} register of the register matrix.
10	Kji3 ($i, j = \overline{1,5}$)	They enable/disable the issuance of the flit stored in the ji^{th} register of the register matrix to the output O_j.
11	K_LOAD_i ($i = \overline{1,5}$)	They enable/disable the transfer of flits from the queue Q_i to the register matrix (HOL blocking).
12	K_LOST_i ($i = \overline{1,5}$)	They enable/disable the reception of incoming flits for the queue Q_i (buffer overflow).

The group controllers RCx and MCx are key elements of the Q-chart as well. While the RCx controls model the supported routing algorithm μ (operating like column-wise random selectors), the MCx elements reflect the implemented row-wise flit scheduling policy. To switch to a different scheduling scheme, the MCx controls need to be reconfigured (internally, this means a different C++ subroutine is selected to manage the element).

Figure 5 shows graphs representing the throughput *versus* the number of inputs/outputs dependencies for the PPIRF switch and the buffered crossbar peers implementing the uniform and the round robin row-wise scheduling policies resulting from our simulation studies based on the Q-chart of Figure 4. Here only low-size switches, with $5 \leq n \leq 25$, are considered (we assumed that the cases $n < 5$ are practically unfeasible for the multiprocessors of the class under consideration).

By analyzing the obtained graphs, it was found that the throughput of the PPIRF switch has the lower bound of approximately 0.78, which was observed to grow smoothly as the number of inputs/outputs of the switch increases. Furthermore, it was found that short-term traffic boosts have no significant effect on switch throughput. Yet, the PPIRF switch was shown to have about 1.2–2.5% higher throughput compared to the buffered crossbar switches controlled by the uniform and the round robin scheduling mechanisms.
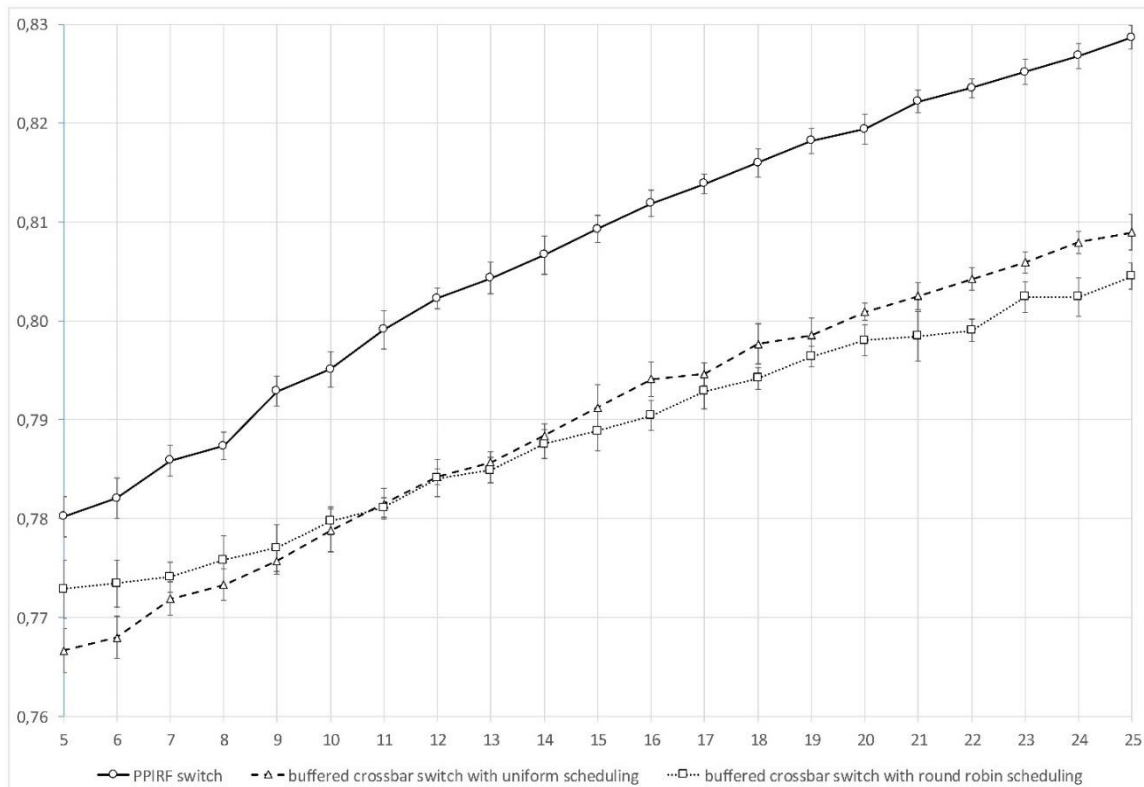
Figure 5. Throughput *versus* the number of inputs/outputs graphs for the low-size PPIRF switch
and the buffered crossbar switches with the uniform and round robin flit scheduling
(the confidence interval is shown for $\alpha = 0.02$).
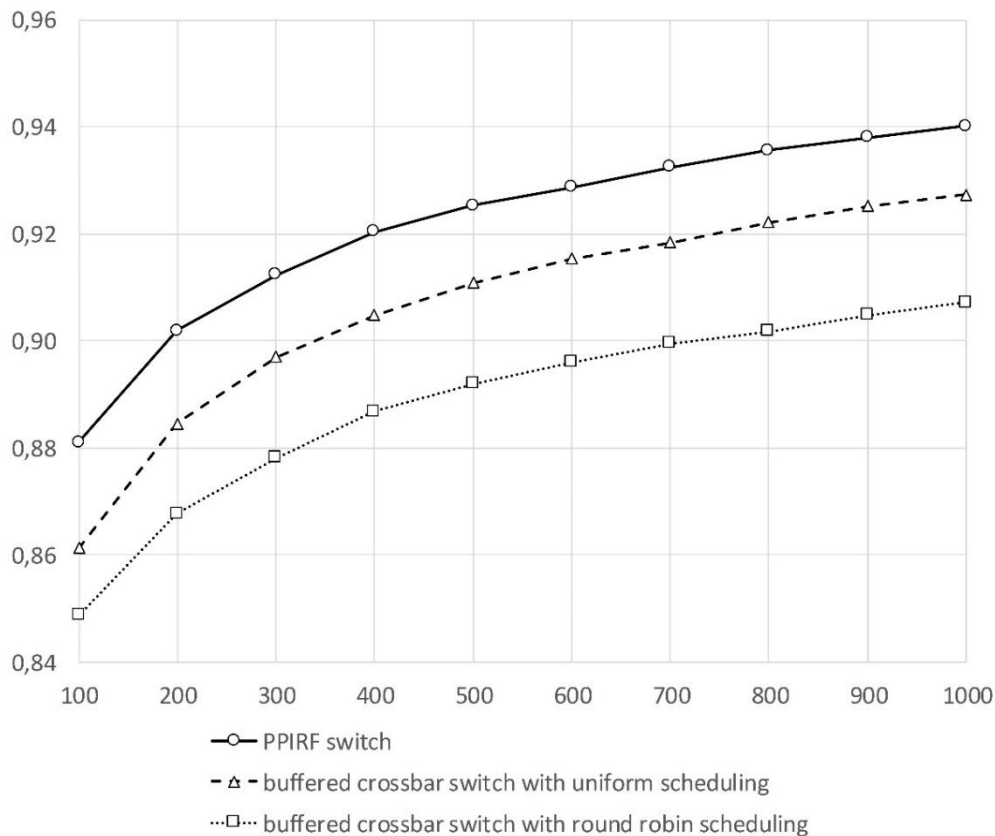

Figure 6. Throughput *vs.* number of inputs/outputs graphs for the high-size PPIRF switch
and the buffered crossbar switches with the uniform and round robin flit scheduling.

We additionally conducted a series of simulation studies considering high-size switches with $100 \leq n \leq 1000$. Figure 6 shows graphs representing the throughput versus the number of inputs/outputs dependency for high-size switches obtained from the simulation studies based on the Q-chart of Figure 4. Since the confidence interval is less than 1% of the throughput average with $\alpha = 0.02$, error bars are not shown in Figure 6. Notwithstanding, it was observed that the throughput of the PPIRF switch exceeds 90% starting at $n \approx 190$; asymptotically, it eventually becomes 100%. This is because for higher values of *n*, the HOL blocking probability P_i^{BL} smoothly decreases to zero. According to Figure 6, the PPIRF switch has approximately 2.5–3.2% and 3.5–3.8% higher throughput compared to the buffered crossbar switches with uniform and round robin flit scheduling, respectively.

Figure 7 presents graphs reflecting the HOL blocking probability *versus* the Bernoulli distribution parameter *p* dependency for PPIRF switches with *n* input/output trunks (the error bars in this figure are not shown, since they are less than 1% of the average). The graphs validate that the probability P_i^{BL} decreases with an increase in the number of inputs/outputs *n*, with the most significant decrease occurring at the maximum intensity of incoming flit streams. For instance, in the case of a 10×10 PPIRF switch, the probability is about 1.62 times lower than for a 5×5 switch. However, for a switch having 15 inputs/outputs, the specified probability is about 1.32 times less. For a 20×20 switch, it decreases by about 1.22 times. Further, for a switch with 25 inputs/outputs, a decline of about 1.17 times was observed. Further simulation study conducted for large-sized switches revealed a further decrease in the probability P_i^{BL}. With $n = 100$, the maximum probability is at approximately 0.04, while with $n = 500$ it is at approximately 0.013 and with $n = 1000$, it is at about 0.0081.
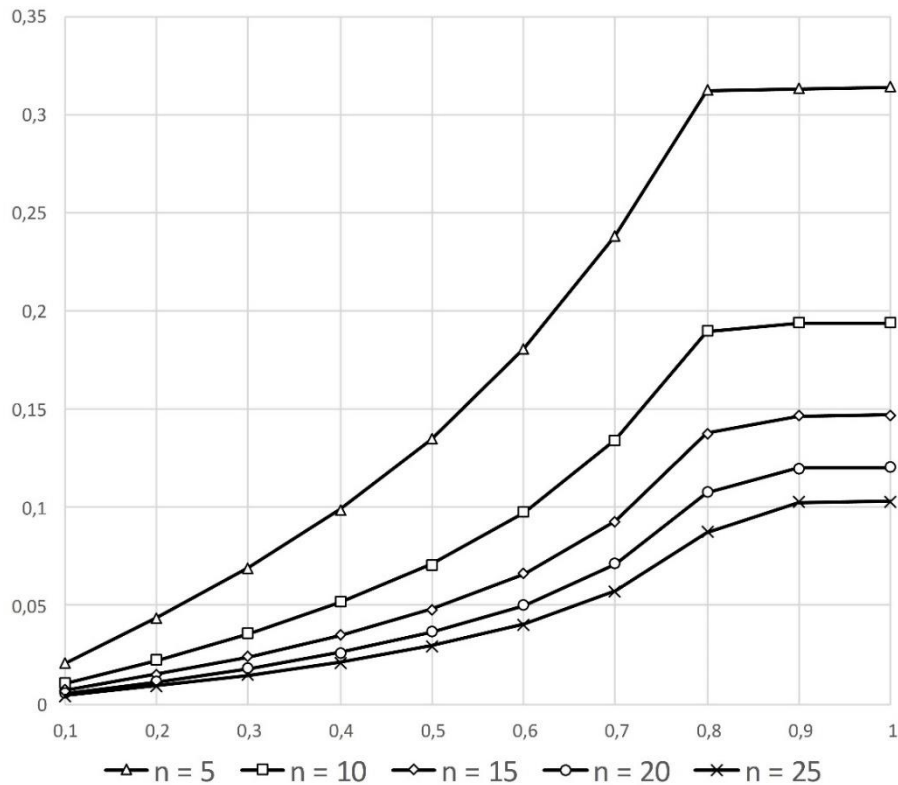

Figure 7. The HOL blocking probability *versus* the Bernoulli distribution parameter graphs for a PPIRF switch having $5 \leq n \leq 25$.

The reason why P_i^{BL} behaves like it is stated above is that the probability (referred to as p_1) for a flit to be transferred to a particular target register of the register matrix decreases as *n* grows; in addition, the decrease is non-linear. For example, while $p_1 = 0.2$ for a 5×5 switch, it goes down to as low as 0.04 for a 25×25 one (if uniform distribution is assumed and $p_1 = 1/n$ therefore). Assuming p_2 to be

the probability that a register of an arbitrary column is not empty (still contains another flit), we obtain $P_i^{\mathrm{BL}} = p_1 p_2$; i.e., $P_i^{\mathrm{BL}} = p_2/n$ in the case of uniform distribution. Hence, P_i^{BL} will go down with n whatever value of p_2 is given. Our simulation has confirmed that p_2 is significantly less than 1 for any admissive traffic patterns.

Based on extra simulation experiments, we studied the delay of the PPIRF switch and compared it to that of the buffered crossbar switches with uniform and round robin flit scheduling (measured in time $n = 25$ slots, with a time slot corresponding to the switch clock pulse period duration). Figure 8 presents the corresponding delay *versus* the Bernoulli distribution parameter p graphs for $n = 5$, $n = 15$ and (the error bars in this figure are not shown for the sake of simplicity). According to the graphs, the proposed architecture has no speed advantage over the peers for $p \le 0.6$; however, as the incoming traffic grows heavier, the PPIRF switch demonstrates better performance in terms of delay. Moreover, the greater the number of input/output trunks, the higher the difference between the compared architectures. Assume $p = 0.8$, which is rather heavy traffic. In this case, a 5×5 buffered crossbar switch with the uniform scheduling policy shows an average delay of as high as 232 time slots, a 5×5 buffered crossbar switch with round robin scheduling is capable to relay a flit in 177 time slots on the average, while a 5×5 PPIRF switch has a delay of as low as approximately 141 time slots. For 15×15 switches, the delay mean values are significantly lower owing to decreased HOL blocking probability and are equal to 84, 89 and 18 time slots, respectively. And finally, for 25×25 devices, the average delays are as low as 29, 34 and 7 time slots, respectively. In general (for any admissive n), the PPIRF architecture is a faster solution than the other switches considered in this study in the case of heavy traffic ($p \ge 0.7$).

5. A SUMMARY OF COMPARISON OF THE PROPOSED ARCHITECTURE TO VOQ-BASED SWITCHES

A comparative study of the PPIRF approach *versus* the latest VOQ-based (including crosspoint-buffered) approaches is of significant interest as well. Some VOQ-based switches are known to be stable; thus, they can achieve up to 100% throughput for any admissible traffic. At the same time, they have a hardware complexity of at least $O(n^2 L)$ (n FIFO buffers of length L at each of n input trunks) and, therefore, the implementation cost may not be acceptable for multiprocessors of the considered class.

Additionally, VOQ-based approaches with no crosspoint buffers require a speedup of about two, which increases the implementation cost and decreases the potentially reachable external speed of the switch. Table 2 presents a comprehensive summary of the comparison results (uniformly distributed traffic is assumed).

6. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we have proposed a parallel pipelined flit switch architecture for mesh-connected multiprocessors with independent flit routing (the PPIRF switch) based on non-VOQ input FIFO buffers and an output register matrix supporting a novel row-wise oldest-flit-first flit scheduling policy. *Via* a comprehensive simulation study, it was found that the proposed approach allows achieving a throughput of at least 78% for practically significant scenarios and up to 100% throughput asymptotically (as $n \to \infty$) with no internal speedup and square hardware complexity in contrast to existing VOQ-based switch architectures. Due to its lower implementation complexity, the PPIRF switch is suitable for large-scale designs and complex network topologies, such as twisted torus, crossed cube, to mention but a few (e.g., see [34]-[35]), requiring many input/output trunks at each node. Compared to known non-VOQ switches with crosspoint buffers, the proposed device demonstrates slightly (1.2–3.8%) higher throughput and substantially (up to several times) shorter delays under heavy traffic patterns.

In the future, it is important to study how to increase the throughput of low-size PPIRF switches by adding some modifications to the scheduling policy to alleviate the influence of the HOL blocking probability and how to efficiently implement our methodology in practical switches taking into account state-of-the-art limitations. In addition, it would be interesting to study the multiprocessor behavior when applying our algorithm to the real communication networks which deliver real-world traffic.

"A Parallel Pipelined Packet Switch Architecture for Mesh-connected Multiprocessors with Independently Routed Flits", J. Al-Azzeh, M. Agmal and I. Zotov.
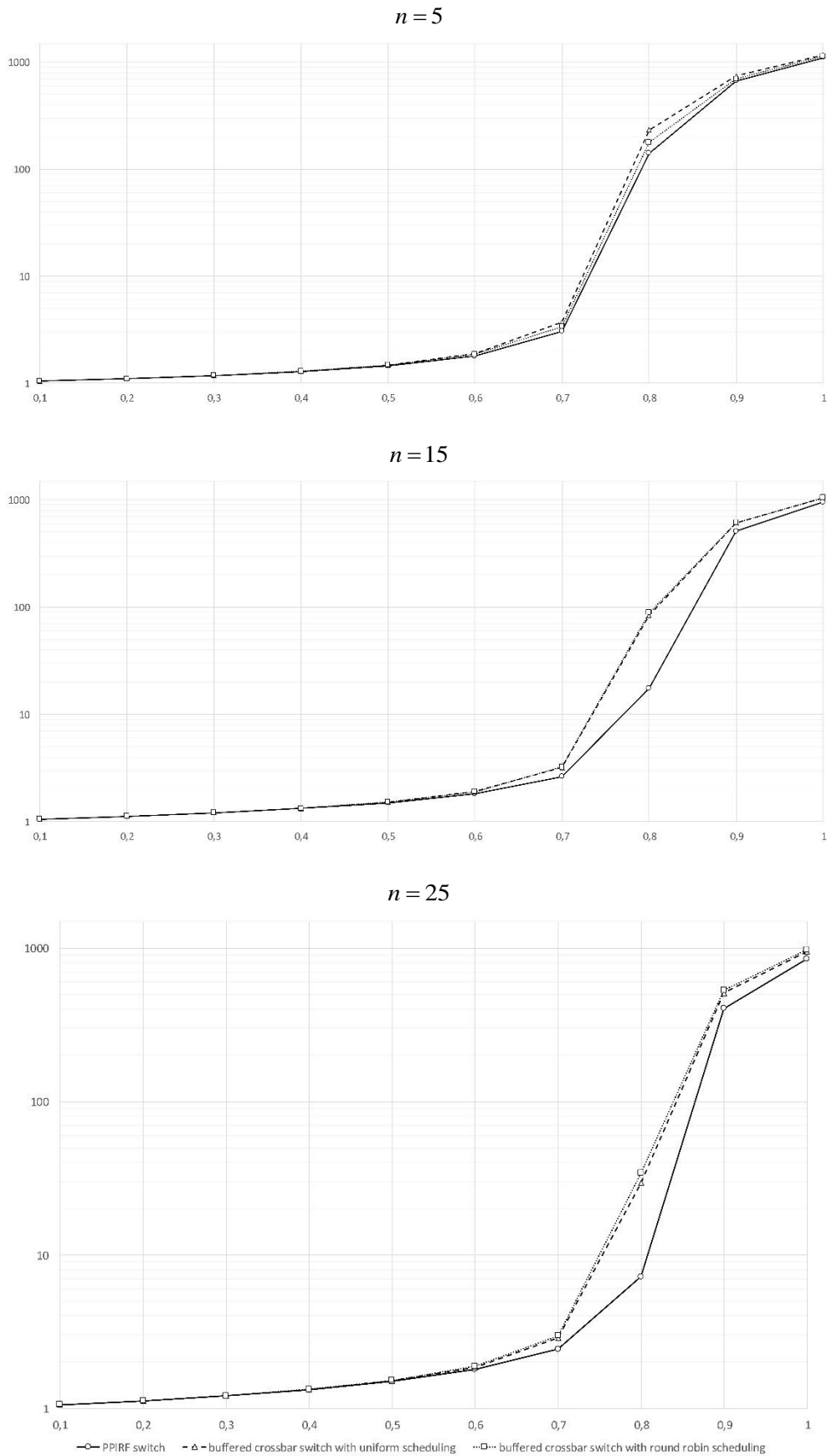
$n = 5$

$n = 15$

$n = 25$

Figure 8. Delay *versus* the Bernoulli distribution parameter graphs for the low-size PPIRF switch and the buffered crossbar switches with the uniform and round robin flit scheduling.

Table 2. Summary of the comparison of the proposed switch architecture with several state-of-the-art architectures.

Switch architecture	Throughput (uniformly distributed traffic)	Hardware complexity	Speedup
iSLIP scheduled VOQ switch [19]	Up to 100% asymptotically	$O\left(n^2 L\right)$	2
SQUID VOQ crosspoint-buffered switch [20]	Up to 100%	$O\left(n^2 L\right)$	1 (not required)
RR/LQF scheduled VOQ switch [24]	Up to 100%	$O\left(n^2 L\right)$	$2 - \dfrac{1}{n}$
Proposed IRF switch architecture	At least 78% for practical scenarios and up to 100% asymptotically	$O\left(\max\left\{n^2, nL\right\}\right)$	1 (not required)

ACKNOWLEDGEMENTS

REFERENCES

[1]   S. Misra and S. Goswami, Network Routing: Fundamentals, Applications and Emerging Technologies, Wiley Telecom, 2014.

[2]   A. A. Jerraya and W. Wolf, Multiprocessor Systems-on-Chips, San Francisco, Elsevier, Inc., 2005.

[3]   Z. Yu, R. Xiao et al., "A 16-Core Processor with Shared-Memory and Message-Passing Communications," IEEE Trans. Circ. Syst. I: Regular Papers, vol. 61, no. 4, pp. 1081-1094, 2014.

[4]   Tilera Corp., "Tile Processor Architecture Overview for The TILE-Gx Series," [Online], Available:
        http://www.mellanox.com/repository/solutions/tile-scm/docs/UG130-ArchOverview-TILE-Gx.pdf, (access date: 30.06.2019).

[5]   A. Olofsson, "Epiphany-V: A 1024 Processor 64-bit RISC System-on-Chip," [Online]: Available:
        https://www.parallella.org/docs/e5_1024core_soc.pdf, (access date: 30.06.2019).

[6]   P. Lotfi-Kamran, M. Modarressi and H. Sarbazi-Azad, "An Efficient Hybrid-Switched Network-on-Chip for Chip Multiprocessors," IEEE Trans. Comput., vol. 65, no. 5, pp. 1656-1662, 2016.

[7]   G. Chen, M. A. Anders et al., "A 340 mV-to-0.9 V 20.2 Tb/s Source-synchronous Hybrid Packet/Circuit-switched 16×16 Network-On-Chip in 22 nm Tri-Gate CMOS," IEEE J. Solid-St. Circ., vol. 50, no. 1, pp. 59-67, 2015.

[8]   A. Mazloumi and M. Modarressi, "A Hybrid Packet/circuit-switched Router to Accelerate Memory Access in NoC-based Chip Multiprocessor," Proc. of Design, Automation and Test in Europe Conference and Exhibition (DATE), pp. 908-911, 2015.

[9]   M. H. Foroozannejad, M. Hashemi et al., "Time-scalable Mapping for Circuit-switched GALS Chip Multiprocessor Platforms," IEEE Trans. Comput.-aided Design of Integr. Circ. and Syst., vol. 33, no. 5, pp. 752-762, 2014.

[10]   M. Karol, M. Hluchyj and S. Morgan, "Input Versus Output Queueing on a Space-Division Packet Switch," IEEE Trans. Commun., vol. 35, no. 12, pp. 1347-1356, 1987.

[11]   L. Deng, W. S. Wong et al., "Delay-constrained Input-queued Switch," IEEE J. Selected Areas Commun., vol. 36, no. 11, pp. 2464-2474, 2018.

[12]   K. Kang, K.-J. Park, L. Sha and Q. Wang, "Design of a Crossbar VOQ Real-time Switch with Clock-

driven Scheduling for a Guaranteed Delay Bound," Real-time Systems, vol. 49, no. 1, pp. 117-135, 2013.

[13] M. J. Neely, E. Modiano and Y. -S. Cheng, "Logarithmic Delay for NxN Packet Switches under the Crossbar Constraint," IEEE/ACM Trans. Networking, vol. 15, no. 3, pp. 657-668, 2007.

[14] S. Durkovic and Z. Cica, "Birkhoff-von Neumann Switch Based on Greedy Scheduling," IEEE Comput. Archit. Letters, vol. 17, no. 1, pp. 13-16, 2018.

[15] C. -S. Chang, D. -S. Lee and C. -Y. Yue, "Providing Guaranteed Rate Services in the Load Balanced Birkhoff-von Neumann Switches," IEEE/ACM Trans. Networking, vol. 14, no. 3, pp. 644-656, 2006.

[16] H. -I. Lee and S. -W. Seo, "Matching Output Queueing with a Multiple Input/Output-queued Switch," IEEE/ACM Trans. Networking, vol. 14, no. 1, pp. 121-132, 2006.

[17] Y. Tamir and G. Frazier, "High Performance Multi-queue Buffers for VLSI Communication Switches," Proc. of the 15th Annu. Symp. Comput. Archit., pp. 343-354, June 1988.

[18] T. Anderson, S. S. Owicki, J. B. Saxe and C. P. Thacker, "High-speed Switch Scheduling for Local-area Networks," ACM Trans. Comput. Syst., vol. 11, no. 4, pp. 319-352, 1993.

[19] N. McKeown, "The iSLIP Scheduling Algorithm for Input-queued Switches," IEEE/ACM Trans. Networking, vol. 7, no. 2, pp. 188-201, April 1999.

[20] Y. Shen, S. S. Panwar and H. J. Chao, "SQUID: A Practical 100% Throughput Scheduler for Crosspoint Buffered Switches," IEEE/ACM Trans. Networking, vol. 18, no. 4, pp. 1119-1131, August 2010.

[21] N. McKeown, V. Anantharam and J. Walrand, "Achieving 100% Throughput in an Input-queued Switch," Proc. of the 15th IEEE INFOCOM, pp. 296-302, San Francisco, CA, USA, Mar. 1996.

[22] J. Chao, "Saturn: A Terabit Packet Switch Using Dual Round Robin," IEEE Commun. Mag., vol. 38, no. 12, pp. 78-84, Dec. 2000.

[23] S. Mneimneh, "Matching from the First Iteration: An Iterative Switching Algorithm for an Input-queued Switch," IEEE/ACM Trans. Networking, vol. 16, no. 1, pp. 206-217, Feb. 2008.

[24] B. Hu, K. L. Yeung, Q. Zhou and C. He, "On Iterative Scheduling for Input-queued Switches with a Speedup of 2-1/N," IEEE/ACM Trans. Networking, vol. 24, no. 6, pp. 3565-3577, 2016.

[25] B. Hu and K. L. Yeung, "Feedback-based Scheduling for Load Balanced Two-Stage Switches,"

IEEE/ACM Trans. Networking, vol. 18, no. 4, pp. 1077-1090, Aug. 2010.

[26] C. -S. Chang, D. -S. Lee and Y. -S. Jou, "Load Balanced Birkhoff-von Neumann Switches, Part I: One-stage Buffering," Comput. Commun., vol. 25, no. 6, pp. 611-622, 2002.

[27] Y. Chen, "Cell Switched Network-on-Chip Candidate for Billion-transistor System-on-Chips," Proc. of IEEE Int'l. Soc. Conf., pp. 57-60, 2006.

[28] A. Olofsson, "Mesh Network," US Patent, no. 8531943 B2, Sep. 10, 2013.

[29] M. Lin and N. McKeown, "The Throughput of a Buffered Crossbar Switch," IEEE Commun. Let., vol. 9, no. 5, pp. 465-467, 2005.

[30] M. Nabeshima, "Performance Evaluation of a Combined Input- and Crosspoint-queued Switch," IEICE

Trans. Commun., vol. E83-B, pp. 737-741, 2000.

[31] I. V. Zotov, "Distributed Virtual Bit-slice Synchronizer: A Scalable Hardware Barrier Mechanism for N-dimensional Meshes," IEEE Trans. Comput., vol. 59, no. 9, pp. 1187-1199, Sep. 2010.

[32] I. V. Zotov et al., "The VisualQChart Simulation Environment," Computer Program Certificate RU 2007611310, appl. 13.02.2007, publ. 27.03.2007.

[33] D. Zydek, H. Selvaraj and L. Gewali, "Synthesis of Processor Allocator for Torus-based Chip Multiprocessors," Proc. of the 7th Int'l. Conf. on Information Technology: New Generations, pp. 13-18, 2010.

[34] A. Samad, M. Q. Rafiq and O. Farooq, "Performance Evaluation of Task Assignment Algorithms in Cube-based Multiprocessor Systems," Proc. of the 1st Int'l. Conf. on Emerging Trends and Applications in Computer Science, pp. 48-51, 2013.

[35]   J. M. Camara, M. Moreto et al., "Twisted Torus Topologies for Enhanced Interconnection Networks," IEEE Trans. Parallel Distrib. Syst., vol. 21, no. 12, pp. 1765-1778, 2010.

[36]   K. Li, Y. Mu, K. Li and G. Min, "Exchanged Crossed Cube: A Novel Interconnection Network for Parallel Computation," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 11, pp. 2211-2219, 2013.

[37]   J. Al Azzeh, "Distributed Mutual Inter-unit Test Method for D-Dimensional Mesh-connected Multiprocessors with Round-Robin Collision Resolution," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 05, no. 01, pp. 1-16, April 2019.

[38]   J. Al Azzeh, "Improved Testability Method for Mesh-connected VLSI Multiprocessors," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 04, no. 02, pp. 116-128, August 2018.

[39]   J. Al Azzeh, "Fault-tolerant Routing in Mesh-connected Multicomputers Based on Majority-operator-produced Transfer Direction Identifiers," Jordan Journal of Electrical Engineering, vol. 03, no. 02, pp. 102-111, April 2017.

ملخص البحث:

فــي هــذه الورقــة، يــتم اقتــراح معماريــة لمفتــاح رُزَم لمعالجــات متعــددة متصــلة بعضــها بــبعض فــي شــكل شــبكة، وذلــك بنــاءً علــى اســتخدام مجموعــة مــن مصــدّات الإدخــال ومصــفوفة تســجيل فــي المخــرج بحيــث يجــري الــتحكم فــي عمــل المفتــاح عــن طريــق مخطــط جدولــة يســتند الــى التوزيــع الزمنــي. يــتم اســتخدام نهــج ثابــت وبســيط مــن أجــل عُبــور المفتــاح، بحيــث يجــري تقســيم كــل رُزمــة الــى مجموعــة مــن الأجــزاء التــي تُوجَّــه فــي مســاراتٍ مســتقلة.   الجهــاز المقتــرح يحقــق نســبة عبــور لا تقــل عــن 78% للعبــور المــوزّع بانتظــام، ويمكــن أن تقــارب فــي حــدّها الأعلــى نســبة 100%. ولــدى مقارنــة الجهــاز المقتــرح فــي هــذه الدراســة بمثيلاتــه مــن مفــاتيح الــرُزَم التقليديــة، وجــدنا أن المفتــاح المقتــرح يصــل الــى نســبة العبــور القصــوى لــه دون الحاجــة الــى زيــادة الســرعة الداخليــة، ناهيــك عــن أنــه أقــل تعقيــداً مــن حيــث المعــدات المســتخدمة. وبالمقارنــة مــع مفــاتيح تقليديــة اســتُخدمت فيهــا آليــات عُبــور نمطيــة، تبــين أن المفتــاح المقتــرح يمتلــك نســبة عبــور أعلــى بقليــل، وأنّ التــأخير فيــه هــو أقــل بشــكل جــوهري فــي بعــض الحــالات ذات الأهميــة العلميــة. فقــد تراوحــت نســبة تفــوق النمــوذج المقتــرح بــين 1.2% و 3.8%، فــي حــين كــان التأخير أقصر بحوالي 5 مرات في بعض الحالات.

الأهداف والمجال

تهدف المجلة الأردنية للحاسوب وتكنولوجيا المعلومات (JJCIT) إلى نشر آخر التطورات في شكل أوراق بحثية أصيلة وبحوث مراجعة في جميع المجالات المتعلقة بالاتصالات وهندسة الحاسوب وتكنولوجيا المعلومات وجعلها متاحة للباحثين في شتى أرجاء العالم. وتركز المجلة على موضوعات تشمل على سبيل المثال لا الحصر: هندسة الحاسوب وشبكات الاتصالات وعلوم الحاسوب ونظم المعلومات وتكنولوجيا المعلومات وتطبيقاتها.

الفهرسة

المجلة الأردنية للحاسوب وتكنولوجيا المعلومات مفهرسة في كل من:

www.jjcit.org   jjcit@psut.edu.jo