논문 16-41-02-09 The Journal of Korean Institute of Communications and Information Sciences '16-02 Vol.41 No.02
http://dx.doi.org/10.7840/kics.2016.41.2.215

이미지센서 네트워크를 위한 강건한 소프트웨어 정의 기법

Robust Software-Defined Scheme for Image Sensor Network

Amirul Islam[◆], Mohammad Arif Hossain[*], Nam-Tuan Le[*],
Chang Hyun Hong[*], Yeong Min Jang[°]

ABSTRACT

Data failure in wireless communications considerably affects the reconstruction quality of transmitted data. Traditionally, fascinating trials have been conducted to overcome the data failure intensifying reliable reconstruction of a media. But, none of these efforts neither effective, computationally inexpensive nor simply configurable to reduce the problems of transmitting media or images. In practice, it is necessary to maintain the quality of transmitted image without sacrificing any data, content, or information. So, to deal with dynamic events such as sensor node participation and departure, during transmission, an efficient scheme is important. For this reason, a new robust scheme has been presented in this paper to minimize the limitation of traditional wireless networking. This scheme uses Software-Defined Image Sensor Network (SD-ISN) to ensure scalability and dependability of the sensor network of handling data losses. Finally, a comparison of our proposed SD-ISN with conventional wireless networking has been presented in simulation to test the robustness and effectiveness of our proposed SD-ISN approach.

Key Words : Software Defined Networking, SD-ISN, OpenFlow, Robustness, PPS

Ⅰ. Introduction

Wireless Sensor Networks (WSNs)[1] have been considered as emerging wireless networks for its wide range of application including distributive services in real time multimedia contents. So it is necessary to ensure the quality of information (QoI) in an acceptable level for the end-user[2,3]. Though many attempts have been deployed to enhance the performance and applicability of WSNs, they left behind some problems. One of the problems of WSNs is that they are stiff to policy changes. Policy changes, related to network factors which have become a key issue due to the ever-changing business or user needs. So it is hard to cope with managing manual reconfiguration or reprogramming of WSN even impossible during any disaster. As a result, it experiences high deployment cost and delay in policy enforcement. Another problem of WSN is low resource reutilization. Different WSNs vendors had deployed their network in the same or overlapping manner without sharing common functionalities, though, it could be achieved by a single versatile network. The third problem is that

※ This work was supported by Institute for Information &communications Technology Promotion (IITP) grant funded by the Korea Government (MSIP) (No. R0127-15-1025, Development of Optical Wireless Communications (OWC) Standardization).
◆ First Author : Kookmin University, School of Electronics Engineering, amirul.0903118@gmail.com, 학생회원
° Corresponding Author :Kookmin University, School of Electronics Engineering, yjang@kookmin.ac.kr, 종신회원
* Kookmin University School of Electronics Engineering
논문번호 : KICS2015-12-412, Received December 28, 2015; Revised February 5, 2016; Accepted February 15, 2016

WSN is very difficult to manage or change the existing program imbedded in the hardware. When a new task has to be imbedded, it becomes very difficult to develop a new network management system and very costly, not efficient, and error prone.

Another extreme considering problem of WSN is contents or packet missing during transmission of any media. This may be because of bit error of transmitting channel and transmitter or receiver errors.

Software-Defined Networks [4] (SDNs) can be an ideal emerging paradigm as a convincing solution of the above problems. SDN is a network architecture that allows the separation of routing intelligence from the underlying devices itself. The separation of data plane and control plane helps network administrators to change the network policies at application needs. In addition, it will help to manipulate the packet forwarding rules by a logically centralized controller. The software-defined sensor network consists of various sensor nodes whose functionalities can be enthusiastically designed by introducing different programs. These also leads to more robustness due to less dependence on the far nodes, less delays, and less energy consumption due to the depreciated communication paths. The rest of this paper has been summarized as follows: Section II introduces the concept of Software-Defined Networking (SDN) with the OpenFlow protocol while proposed architecture has been illustrated in Section III. And the simulation results has been shown in section IV. Finally, Section V concludes the paper.

Ⅱ. Introduction to Software-Defined Networking

Software-defined networking is comprehensive networking technology[4-7] that can be employed to a variety of approaches. Conventionally, IP packet based networks consist of nodes running with limited distributed protocols to route packets. This packet forwarding is maintained by some controlled distributed algorithm to perform routing decisions.

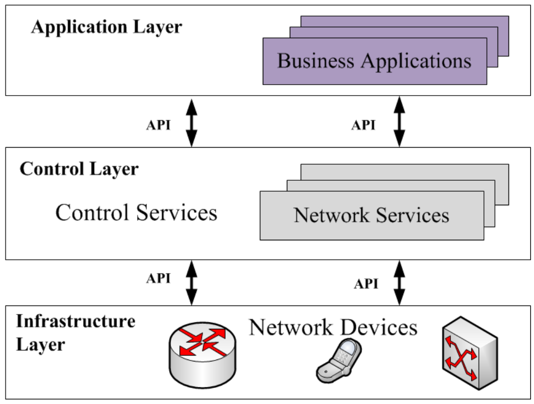


Fig. 1. Architecture of Software Defined Networking.

But in these protocols, there has no control over of forwarding tables at any level at user requirements.

Even if these algorithms cannot satisfy the aforementioned problems of WSNs. Nevertheless, we reckon that it is necessary to introduce the fundamental idea of SDN into WSN for image transmission and adapt it into a viable approach to solve WSN-inherent difficulties. Admittedly, we visualize SD-ISN to reconstruct traditional WSN into arrangements that signify:

- Versatility: having the utility in multiple applications, sensors have become user-independent. These can be realized by decoupling (i) the data plane that virtually holds all packet forwarding rules, and (ii) the control plane that provides a global view of the network to the applications.
- Resiliency: simple to handle policy changes. This can be realized by the centralized and notably flexible network control in SD-ISN, where the resiliency can be satisfied by the fully user-customizable flow tables.
- Manageability: easy to manage by introducing another application or interface over the control plane such as open APIs. The global network view from the application plane is an excellent functionality, because the users like to see the centralized view in real time.

Meanwhile, the aforementioned situation of Software Defined Networking (SDN) depicts a highly encouraging standard to render the network including an effective level of versatility and scalability. Firstly, it reveals the plumb integration by separating the networking control from the underlying devices (e. g.; switches, routers) that advance the traffic. Secondly, with the simplification of the network and separation of the data planes and control planes, the control plane is realized in a network operating system that is logically centralized and the data planes have become simple forwarding devices. Figure 1 provides an illustration of the basic SDN architecture. The separation of the control plane and the data plane can be understood by introducing a well-defined interface between the data plane and the controller. The controller runs over the physical devices of the data planes through OpenFlow protocol, a well-defined programming interface.

OpenFlow protocol maintained by ONF[8], the first leading authorized communications interface linking between the forwarding and controls layers of an SDN architecture allow manipulation and control of the forwarding plane of network devices (e. g.; switches and routers) both physically and virtually.

Several SDN solutions have manipulated few restricted protocols such as Cisco's Open Network Environment Platform Kit[9] (onePK), Juniper's contrail[10], and other opportunities that comprise the Forwarding and Control Element Separation (ForCES) framework[11]. OpenFlow helps SDN architecture to adapt the high-bandwidth, dynamic nature of user's applications, adjust the network to the different business needs, and interestingly, reduce management and maintenance complexity. However, OpenFlow will be ambiguous and may have particular interpretations, which may grant implementation freedom to vendors.

The algorithm of the OpenFlow protocol for conventional and our proposed software-defined networking have been illustrated in figure 2. When a new flow or packet reaches, some lookup manner originates in the primary lookup table. When the packets do not acknowledge what to do with a distinct incoming packet, default information to forward the packet to the controller is ''send to controller''. While in the conventional case, the packets or flows are dropped if not matched. Figure 3 describe the functionalities of an SDN controller. It can be classified into four categories: a high-level language to define network operation policies; a rule update process to install rules from those policies; a network status collection process to gather network infrastructure information; a network status synchronization process to provide a global view of the whole network using network status collected by each individual controller.

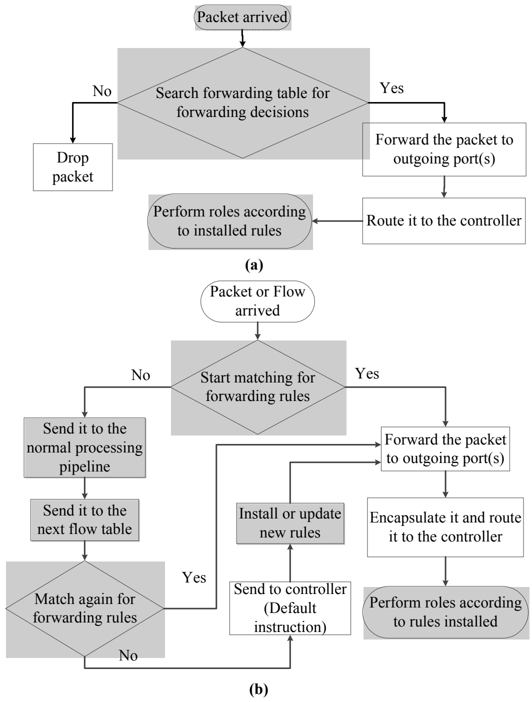


Fig. 2. Algorithm for (a) Conventional IP-based Network (b) Proposed Software Defined Networking

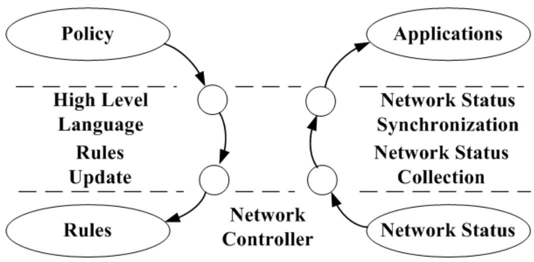


Fig. 3. Functionalities of SDN controller

Ⅲ. Proposed Architecture

We have proposed a Software-Defined Image Sensor Network (SD-ISN) to compensate the problems of wireless sensor networks. It is an architecture featuring a distinct separation between a data plane and a control plane, and a robust OpenFlow protocol between the two planes. The data plane consists of reconfigurable physical devices (e. g.; sensors, switches, routers) performing flow-based packet forwarding, and the control plane consists of one or more controller that logically centralizes the network intelligence, performing network control such as routing, forwarding, and QoS control. The proposed architecture has been depicted in Figure 4. The whole idea of this architecture is to make the underlying network programmable by manipulating a user-friendly flow table. It combines the whole sensor nodes into some cluster nodes that finally combines in few cluster heads. It also selects the associated cluster head or nodes for any new incoming packets.

SDN architecture has the flexibility to centralize few or all parts of the networking devices at a more robust node. Fairly, OpenFlow, forwarding, and many routing decisions may be carried out by the individual sensor nodes. Nevertheless, a centralized controller can take long-term decisions, such as which protocol to use, which way to route the packet, and which parameters to select. A central controller has also sufficient information about application requirements and limitations, which can also produce a consequence as well. An important consequence of the SD-ISN principles is the separation of concerns induced between the definition of network policies, their involvement in reconfigurable switching devices, and the forwarding of packets or traffic. This separation is the key issue to the desired flexibility, breaking the networking control problem into pieces, making it easier to the user, and simplifying network operation and innovation.

The central controller (SDN controller) needs to discover the actual topology and the quality of the links. What is best depends on how it is supposed to be used. However, this configuration should be reconfigurable and re-discoverable. The controller should initiate packet flow when the applications demand any information, otherwise the lifetime of the controller will reduce. A packet holds specific information about how a packet behaves over a link. This information can be employed in a simulator with a method which can be described as OpenFlow-based simulation. Figure 5 represent the simulation architecture for the proposed SD-ISN scheme. It mainly consists of sensor nodes based packets forwarding technique. Here, the sink is the most powerful node, which is typically not equipped with sensing units, but it has a powerful processing capability and is outfitted with multiple sensor nodes. The whole area is divided into several section (clusters). Each section consists of several sensor

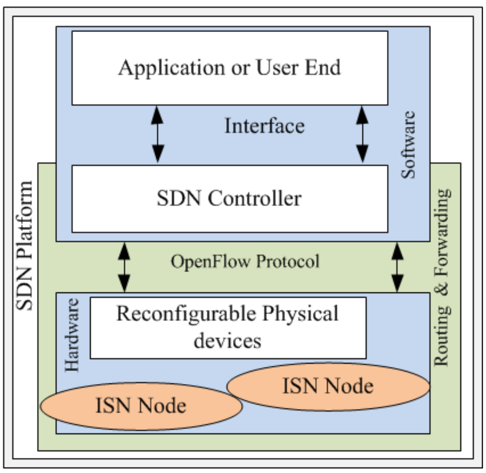


Fig. 4. Proposed overall Software-Defined Image Sensor Network (SD-ISN) Architecture

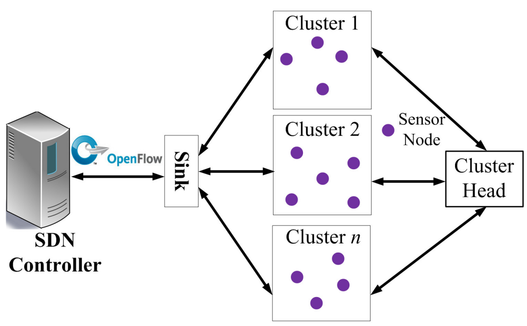


Fig. 5. Proposed SD-ISN Scenario for Simulation.

nodes. The sensor nodes have been considered as the packets of any task initiated from the user-end. The received tasks from the applications are controlled by the SDN controller and OpenFlow protocol to originate the required rules on the sensor nodes or cluster head. The cluster-heads forward the alert messages (packets) to the controller in case of uneven event detection, which contain information about the reported events. This simulation technique using this architecture and the algorithm in fig. 3(b) has been used for execution of the simulation.

Ⅳ. Simulation Results

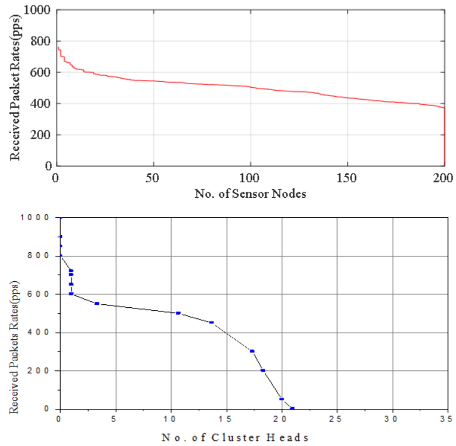
This section represents the simulation results of the proposed software defined image sensor network (SD-ISN) architecture using the algorithm of fig. 3(b) and the simulation architecture of fig. 5. The results show a comparison between the conventional wireless networking and our proposed SD-ISN approach which is shown in figure 6. The parameters used for simulation has been presented in table 1 and the comparison of the results has been shown in table 2. This result represents how we can compensate the problem of wireless sensor network using SD-ISN. For the execution of the our proposed algorithm, we consider 200 image sensor nodes and 35 cluster heads (sum of clusters) although users can consider any numbers at their requirements or needs. We have used 200 sensor nodes for simplify the simulation and also used 35

Table 1. Parameters for simulation purpose

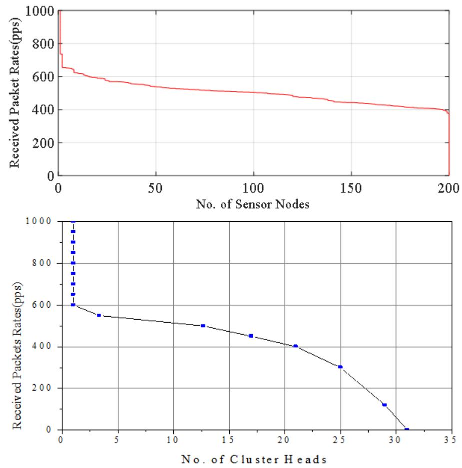
Maximum Packets flow (pps)	Image Sensor Nodes	Cluster Heads
1000	200	35

Table 2. Comparison of simulation results

Approach	Received Packet Rates (pps)		Cluster Heads
	Maximum	Minimum	
SD-ISN	1000	400	31
Conventional IP Network	800	380	22



a) Conventional Wireless Networking Approach



b) Software-Defined Image Sensor Networking Approach

Fig. 6. Simulation Results of Proposed SD-ISN Architecture

cluster heads because in real-time applications a center can accomodate around 5 sensor nodes. The total packets were operating in the network have been assumed 1000 packets per sec. (pps) for both cases. In the upper part of figure 6(a) shows that the maximum packets received rates is 780 pps and minimum is 380 pps, when the maximum number of nodes is operating. While, for our proposed SD-ISN, received packet rates is 1000 pps and minimum is 400 pps. The other part shows the characteristic of the received packet rates as a function of a number of cluster heads acting. For conventional networking, the maximum received packet rates are 800 pps and for SD-ISN 1000 pps and the cluster heads in action are 31 cluster heads and 22 cluster heads for SD-ISN and traditional networking respectively, when the received rate is zero. This also shows that the traditional networking has less cluster head in action than our proposed SD-ISN scheme and can receive less packet even if at the initial stage. This represents that our proposed scheme transmit more packets efficiently to the sensor nodes (e. g.; cluster head) as well as reduce the amount of dead nodes. For the robustness of applications, efficiency and the reduction of the delay are important in data processing of sensor networking which is satisfied by our proposed SD-ISN scheme.

Ⅴ. Conclusion

In this paper, we have presented a robust approach of software-defined image sensor networking to reduce the packet losses, data failures during the transmission of image. We have presented a comparison between conventional wireless and our proposed SD-ISN method to show the packet losses, dead nodes or data losses for the transmission of image. The reduction of data losses or failure can be a very considering factor in case of networking. Finally, our proposed SD-ISN method proves its effectiveness, scalability, and robustness in wireless image sensor networks.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393-422, 2002.

[2] S. Kim, C. Kim, H. Cho, T. Yang, and S.-H. Kim, "Symmetric inter-communication scheme among mobile objects in wireless sensor networks," *J. Korean Inst. Commun. Inf. Sci.*, vol. 40, no. 10, pp. 2014-2025, Oct. 2015.

[3] C. Kim, K. Jung, and S.-H. Kim, "Opportunistic multipath routing scheme for guaranteeing end-to-end reliability in large-scale wireless sensor networks," *J. KICS*, vol. 40, no. 10, pp. 2026-2034, Oct. 2015.

[4] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A survey on software-defined networking," *IEEE Commun. Surveys & Tuts.*, vol. 17, no. 1, pp. 27-51, 2015.

[5] D. Kreutz, F. M. V. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," in *Proc. IEEE*, vol. 103, no. 1, pp. 14-76, Jan. 2015.

[6] Y. Jarraya, T. Madi, and M. Debbabi, "A survey and a layered taxonomy of software-defined networking," *IEEE Commun. Survey & Tuts.*, vol. 16, no. 4, pp. 1955-1980, 2014.

[7] N. T. Le, M. A. Hossain, N. Saha, Md. S. Ifthekhar, T. Nguyen, C. H. Hong, and Y. M. Jang, "Software-defined networking architecture for energy balance sensor network," in *Proc. KICS Winter Conf.*, vol. 5, no. 2, pp. 121-123, Jun. 2015.

[8] *OpenFlow switch specification*, version 1.4.0 (Wire Protocol 0x05), Oct. 2013.

[9] CISCO, *Cisco's One Platform Kit (onePK)*, from http://www.cisco.com/en/US/prod/iosswrel/onepk.html

[10] Juniper Networks, Contrail, *A SDN Solution Purpose-Built for the Cloud*, from http://www.juniper.net/us/en/productsservices/sdn/contrail/

[11] A. Doria, J. Hadi Salim, R. Haas, H. Khosravi, W. Wang, L. Dong, R. Gopal, and J. Halpernand, *Forwarding and control element separation (ForCES) protocol specification*, RFC 5810, pp. 2070-1721, Mar. 2010, from http://tools.ietf.org/html/rfc-5810/

Amirul Islam

2014년 9월 : Khulna University of Engineering and Technology in Electrical and Electronic Engineering 졸업

2015년 9월~현재 : 국민대학교 전자공학과 석사과정

<관심분야> Localization, SDN, Optical and Camera Communications

Mohammad Arif Hossain

2014년 9월 : Khulna University of Engineering and Technology in Electrical and Electronic Engineering 졸업

2014년 9월~현재 : 국민대학교 전자공학과 석사과정

<관심분야> Image Processing, Computer Vision, Optical Wireless Communication (OWC), Digital Signage, Internet of Things (IoT)

Nam-Tuan Le

2005년 2월 : HaNoi University of Science and Technology 졸업

2011년 8월 : 국민대학교 전자공학과 석사

2014년 2월 : 국민대학교 전자공학과 박사과정

2014년 3월~현재 : 국민대학교 조교수

<관심분야> Sensor Network, Visible Light Communication

Chang Hyun Hong

2015년 2월 : 국민대학교 전자공학부 졸업

2015년 3월~현재 : 국민대학교 전자공학과 석사과정

<관심분야> Optical Wireless Communication, Image processing

Yeong Min Jang

1985년 2월 : 경북대학교 전자공학부 학사

1997년 2월 : 경북대학교 전자공학부 석사

1999년 8월 : Univ. of Massachusetts Computer Science 박사

2002년 3월~현재 : 국민대학교 교수

<관심분야> OWC, WPAN, IoT, 5G Mobile communications