

# An Automatic Online Disaster Monitoring Network: Network Architecture and a Case Study Monitoring Slope Stability

<https://doi.org/10.3991/ijoe.v14i03.7669>

Jianming Wang<sup>(✉)</sup>, Xiaoqin Yang  
Taiyuan University of Technology, Taiyuan, P. R. China  
8844.4321@163.com

**Abstract**—The development of geosensor networks (GSNs) is important for advancing the state of the art of disaster monitoring. An online automatic unattended disaster monitoring system can reduce or prevent geologic disasters and thereby protect life and property. Currently, most GSNs are independent and in most cases serve only one area. However, a GSN can be much larger and more complex and a modern GSN is a heterogeneous wireless sensor network. This paper proposes a novel GSN disaster monitoring architecture. In this architecture, four layers are used to collect, manage, transport, and process the observational data. The data server layer applies the Open Geospatial Consortium Sensor Web Enablement standards to integrate and share heterogeneous monitoring data. Sensor metadata and observation data are packaged into a virtual sensor that is transported from the data layer to the application layer using Sensor Observation Service. To demonstrate the applicability of this architecture, we developed and deployed the Mine Slope Monitoring System to monitor the safety and stability of a slope in an open-pit mine in China. This paper outlines the architecture and describes the four layers from which the GSN is built. The paper concluded by describing this system and its use at the Chinese mine.

**Keywords**—Geosensor networks, Disaster monitoring, Sensor web, Geological disasters

## 1 Introduction

In recent decades, as global climate anomalies increase, the number of natural disasters is rising and environmental disasters due to human activity are also increasing. These disasters commonly cause enormous damage to human society and are a serious threat to people's lives, property, and safety. We should respond to these disasters, taking many different measures to prevent them and to try to reduce their effect on people's lives. Modern geosensor networks (GSNs) support many important applications including global environmental monitoring, natural disaster prediction, and critical infrastructures protection [1]. A GSN is a distributed ad hoc wireless network of sensor-enabled miniature computing platforms (a sensor network) that monitors environmental phenomena or geologic processes in a geographic space [2,3]. There

are a number of instances where GSNs have been used successfully for disaster monitoring. For example, GSNs are commonly deployed in remote, inaccessible, and hazardous areas such as volcanos [4,5] and used to monitor hazardous phenomena like debris flows [6] or floods [7]. In 2015, we designed the architecture for a GSN-based disaster monitoring system in a mine [8] that gave a comprehensive view on the important technologies for ground surface monitoring systems.

Some of these monitoring systems have attained automatic monitoring so that they can continuously monitor the environment in real-time or near-real-time without human intervention. However, in most cases these systems can only service their local departments or communities; their services are independent and self-contained. As a result, these systems cannot share data between communities and cannot realize the benefits of interoperability. In some cases, if the monitoring data from different sensor networks can be processed by data fusion, the result is more substantial. In addition, other communities or users want to obtain these valuable data for disaster research, so there is a need to disseminate real-time public observational data from sensor networks to other users. Under certain conditions, sensor resources and sensor system also need to be shared between different departments or communities. The appearance and development of sensor webs allows new methods for sensor sharing and interoperability to be implemented. The Sensor Web Enablement (SWE) initiative of the Open Geospatial Consortium (OGC) aims to standardize sensor webs with a suite of standardized interfaces [9]. Sensor web technologies play an important role in disaster management and environmental monitoring. The SWE technologies can provide real-time or near-real-time geospatial data to support timely decision-making [10].

In most cases, GSNs need to collect data either periodically or continuously. To solve the problem of processing large amounts of complex observational and monitoring data, we presented methods for using raw sensor data preprocessing middleware in a data management center to compress the data to facilitate data transmission and management. In addition, we enabled data sharing and remote publication with the support of a sensor web service interface. To fulfill these objectives, the main contribution of this paper is a geosensor network service architecture adapted to support in-situ monitoring. To demonstrate our work, a mine disaster monitoring services system based on that architecture is described. This system is an automatic online GSN system.

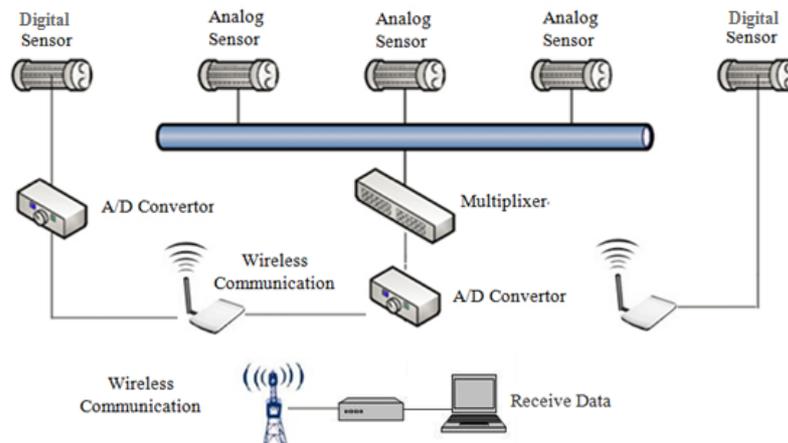
This article is divided into five sections. Section 1 is this introduction. Section 2 provides an overview of the related work and an introduction to relevant concepts such as GSNs and sensor webs. Section 3 describes the architecture in detail and presents the four service layers. Section 4 describes the mine disaster monitoring system as a case study and demonstrates the work. Finally, Section 5 summarizes the methods presented.

## 2 Background

In this section, we briefly explain the related technologies used for disaster monitoring systems; GSN, OGC, and SWE.

### 2.1 Geosensor Networks

In the last 10 years, wireless sensor networks (WSN's) have influenced the field of geosciences in significant ways. GSN technology, which is today still largely in the research stage, will add significant new capabilities to modern geoscience. The GSNs currently I use are specialized applications of WSN technology in geographic space that detect, monitor, and track environmental phenomena and processes [11]. Compared with traditional monitoring methods, automated GSNs can reduce the labor costs associated with field surveys and provide real-time or near-real-time observations and automatic records [12]. Automatic online GSN systems are the current trend for regional disaster monitoring. The units that constitute a basic automated disaster monitoring system are shown in Figure 1.



**Fig. 1.** Diagram illustrating the three basic components that constitute a typical automated monitoring system. Those components are made up of sensor resources, wireless communication resources, and data receiving resources

Environmental sensors are increasingly smart, small, and energy efficient. In remote, sensitive, or hazardous environments and with very high spatial and temporal granularity, GSNs are much better than conventional monitoring techniques. In the near future, GSNs will be composed of thousands or millions of geosensors. Additional information on the future of GSNs is available from sources like [2] and [11]. With the rapid development of sensor networks, more and more GSN systems faced with complex application backgrounds are emerging. This has presented new research opportunities and technological challenges to overcome, such as distributed pro-

cessing, sensor data integration and management, the approach known as modeling, monitoring, detecting environmental processes, and the development of algorithms for decentralized spatial computation. For the problems presented in this paper, the main challenges are distributed processing and the integration of vast amounts of heterogeneous sensor data in real time. The ideal solution is to create a sensor web with real-time access and share sensor data in a uniform way similar to the way data are shared on the Internet. To do this, standardized sensor service interfaces are required.

## **2.2 Sensor Webs**

Sharing, accessing, and integrating diverse sensors into observation systems is not easy because there are a plethora of sensor manufactures and many different communication protocols. A coherent infrastructure is needed to handle sensors in an interoperable, platform-independent, and uniform way. The concept of the sensor web reflects such an infrastructure for finding, sharing, and accessing sensors and their data across different applications. The sensor web hides the heterogeneous sensor hardware and communication protocols from the applications built on top of them [13]. A sensor web is a kind of infrastructure that allows users to automatically access and extract more and different sensor observation data from the Internet. The OGC's SWE presents a reusable, scalable, extensible, and interoperable service oriented sensor web architecture [14]. The SWE standards enable developers to make all types of sensors and sensor data repositories discoverable, accessible, and usable via the web. In SWE, there are standardized information and service models.

The SWE technical specifications are not described in this paper; the interested reader is referred to the OGC specification documents on the Open Geospatial Consortium web site, <http://www.ogcnetwork.net/swe>. SWE consists of two standardized information models. They are 1) the Observations and Measurements (O&M) [15] standard defines a domain independent, conceptual model for the representation of spatiotemporal measurement data. The purpose of the O&M is to alleviate or even remove the need for sensor-specific formats for describing the data retrieved from the sensor networks. 2) The Sensor Model Language (SensorML)[16] for the description of sensor metadata. SensorML specifies a model and XML encoding for the description of all kinds of sensor related processes. In SensorML a sensor is defined as a process that is capable of observing a phenomenon and returning an observed value. It allows a detailed description of a process including a listing of its inputs, outputs, parameters, and process methods. Further metadata about a process can also be defined including its identification and classification, as well as characteristics such as its temporal availability or its spatial description. The SWE service model is composed of standards that specify the interfaces of the different Sensor Web services. The Sensor Observation Service (SOS) is used to manage the configured sensors and requests for sensor data, especially observations. The SOS is the intermediary between a client and an observation repository or near real-time sensor channel. A client can also get information about the sensors and sensor platforms by requesting a SOS instance.

Some sensors or sensor platforms support dynamic configuration at runtime. This can be, for example, the protocol for the sampling rate or the steering of a movable sensor platform. Tasking of sensors in an interoperable way can be done by using the Sensor Planning Service (SPS). The SPS is a web service interface that allows clients to submit tasks to sensors for tasking sensors and setting their parameters [13,17]. The OGC Sensor Alert Service (SAS) allows consumers to subscribe to sensor data with some filter criteria such as a bounding box or a simple measurement value threshold. The OGC Web Notification Service (WNS) is able to receive notifications and to forward them to registered clients via different protocols such as email or SMS. This way one can ensure that important notifications reach their destination as soon as possible. The SAS and WNS were revised and enhanced to form a Sensor Event Service (SES) specification during the development of the SWE 2.0 standards.

### 3 Overall Architecture of the Disaster Monitoring System

The disaster monitoring systems must survive extreme environmental and weather conditions and operate for long periods with limited human intervention, thus energy harvesting and energy efficiency are major design considerations. Near-real-time data collection is another important feature that allows the system to support time-sensitive environmental studies. This requirement necessitates convenient yet reliable long-haul wireless communications links [18]. Therefore, we propose that GSN service architectures for disaster monitoring systems should consist of a data sensing layer, a data management layer, sensor services, and an application layer. The detailed specifications of these four layers are described below. A summary of the architecture is illustrated in Figure 2.

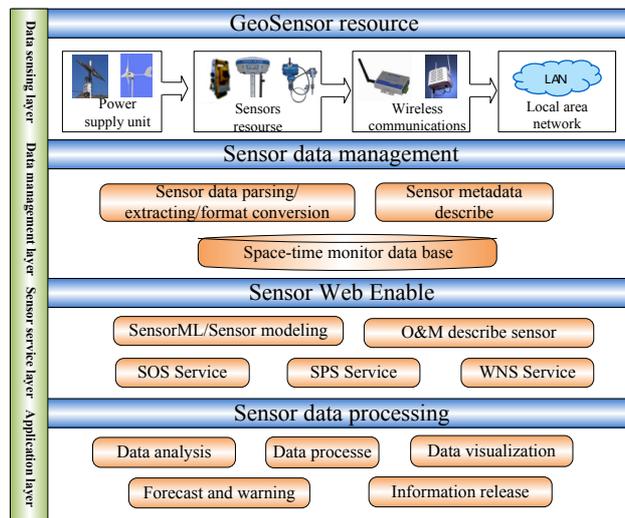


Fig. 2. Overall view of the four-layer architecture of the disaster monitoring system

### **3.1 Networks Data sensing layer**

This layer contains continuous or periodic observation data from sensors in the monitored area. Sensor resources include power sources. The different sensors and the communications system are the core of the sensor network and the front end of the monitoring system. The types of sensors and their roles depend on what is being monitored. Sensor resources for monitoring may be stationary or in motion and can gather data in an in-situ or remote manner [1]. Batteries are the main power sources for sensor nodes. If a sensor requires more power than batteries can supply, other power sources that harvests power from the environment such as solar panels or wind-powered generators may be added to the node depending on the application and the type of sensor used. Sensor location must also be taken into account when considering alternative power sources. Environmental monitoring commonly requires continuous real-time or near-real-time observation. The data from continuous observations can be transferred to the local area network (LAN) by wireless communication (e.g., GPRS and CDMA, two common mobile phone standards).

### **3.2 Data management layer**

The data for environmental monitoring is time-series data. In most cases, each sensor in the sensor network produces streaming data and the data are also commonly collected in real-time so the amount of data increases over time. Generally there are two data types. One is static data, such as data that describe the sensor's characteristics or condition (sensor metadata); the other is dynamic data, the data the sensor perceives or "sees" [19]. It is imperative to effectively manage these data by building a data management system for service and computing to provide adequate protection and support. A distributed data management system combined with sensor web technology is an ideal solution for data management in a GSN. The data management system also performs data pre-processing. Before the data are stored, it commonly needs parsing or to be extracted from the raw sensor data. In some cases, format conversion is required. In addition, data management systems must have robust outlier and error handling capabilities to provide users with reliable data.

### **3.3 Sensor service layer**

The goal of the sensor web research field in the service layer is to put sensor resources on the web and make them available to applications. As long as the data management system server has access to the Internet, the rest of the work falls to SWE. SensorML is used to describe sensor measurement models with O&M standard coding and register sensors on the Sensor Web Registration Center. The Sensor Web Registration Center is the channel between service providers and clients. It provides a specific mechanism by which service providers and clients can find each other [20].

When the SOS request from a web client is sent to the sensor registration center, the observation data from the data management system can be conveyed to any user on the Internet, thus achieving data sharing and interoperability. The SPS controls

monitoring sensors to assist in data collection. Assistance from the SPS may be required when some specific event occurs (e.g., rainfall), when one or more sensor(s) start an observation, or when it is necessary to control the frequency of data return. The SAS can cause the disaster monitoring system to announce an alert message when a data processing result exceeds a preset threshold.

### **3.4 Application layer**

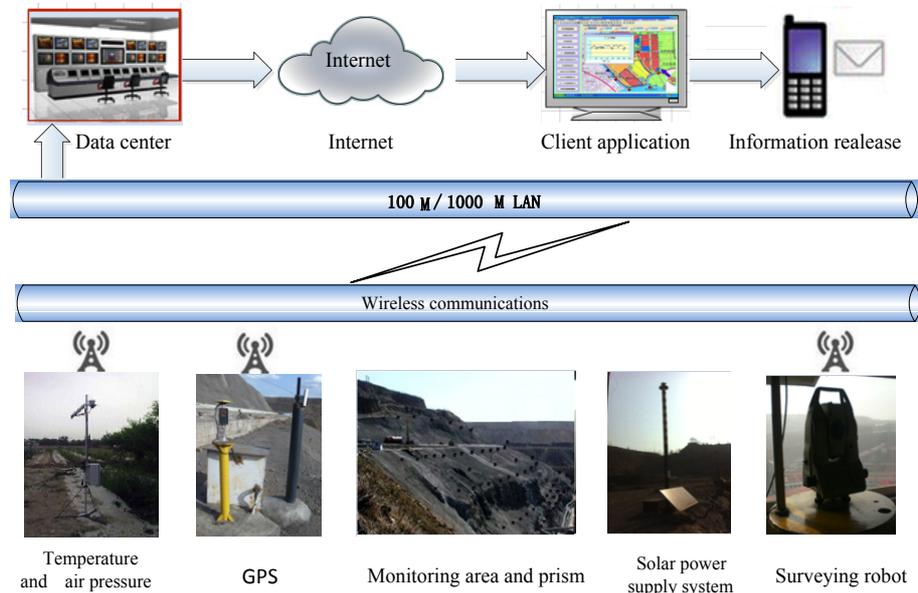
The client application is the core of the disaster monitoring systems. According to the actual need, applications can customize web or desktop clients. The client application is responsible for processing and analyzing data from the data management system. Early warnings about potential disasters may be predicted, forecast, or issued by processing and analyzing the data collected with the support of the client application and SWE services. To realize the automatic publication of early warning information, a SAS should be subscribed to the client. The sensor data are continuously published to the SAS and if a specified filter criterion is matched, the SAS forwards the data to the client application. The client application can also register for alarms if certain events occur. In those cases, the SAS triggers a WNS to notify the client via a defined communication protocol. For example, a user can receive an alert via email or Simple Message Service (SMS) if the water level at a gauging station is above a specified height.

## **4 Case Study: A Slope Stability Disaster Monitoring System**

Many long-lived open-pit mines have transitioned to underground mining or are about to transitioned to underground mining. The risk of disasters is increasing. Therefore, management should pay a great deal of attention to uniform disaster control [21]. Geological disaster monitoring technology for mines has entered a new era. Many modern mines have developed numerous kinds of sensor network monitoring systems. In most cases, however, these monitoring systems are only semi-automatic, independent, and non-relative system that only provide service for one type of event. The sensor data are not shared between systems. If some geological hazard data are integrated with data from other sensor systems and the data are processed together, the results can be more useful. We developed and deployed an integrated mine slope monitoring system (called MSMS) based on a GSNs and SWE. The monitored area is a typical slope at an opencast coal mine. The MSMS is responsible for monitoring slope stability. If the slope collapses or slides, it will disrupt underground mining and cause an accident.

### **4.1 Framework of the MSMS**

The MSMS framework is based on the architecture described in Section 3. Figure 3 shows the MSMS framework; it consists of the elements described below.



**Fig. 3.** The framework of the MSMS and photographs of examples of its components

There are three types of sensors in the GSN of the MSMS. These sensors are a surveying robot (SR), GPS devices, and an air temperature and pressure sensor. The GSN can collect near-real-time data from the sensors in the monitoring region and is able to detect some anomalies before the slope destabilizes. The SR (also called an auto or robotic total station), is an integrated system that can automatically recognize and track targets, measure vertical and horizontal angles, measure distances, and record data. Its deficiencies are that it is limited to line-of-sight measurements and has a limited surveying scope. A number of monitoring points are laid out on the terrain. Each monitoring point has a fixed reflecting prism as a sighting mark (target) for the SR. The SR is in a fixed position in the monitoring area, so its coordinates are affected if the slope on which it is situated becomes unstable. Another sensor system in the monitoring area is composed of six GPS receivers. Two of the six are used to update the SR's coordinates. The other four monitor blind areas and make up for the line-of-sight limitation of the SR. The GPS receivers can monitor continuously 24 hours per day. The SR monitors periodically and observes once every half hour but when it starts to rain, the SR begins to collect data continuously. A single temperature and pressure sensor is used to obtain the temperature and pressure in the monitoring area for the SR.

#### 4.2 Data Transmission

After the GSN is established and each sensor node can collect data, the sensors need to transmit the data using a transmission protocol. This is commonly a wired serial connection, such as RS-232, but can also be a LAN or even a wireless connection.

tion (e.g., GPRS, Bluetooth). In this MSMS, we have equipped a data center to receive observation from the GSN via wireless communications systems. A database server and a web server are housed in the data center and any remote user can query the SOS to obtain data from the database through the Internet. Figure 4 illustrates the sensor data stream.

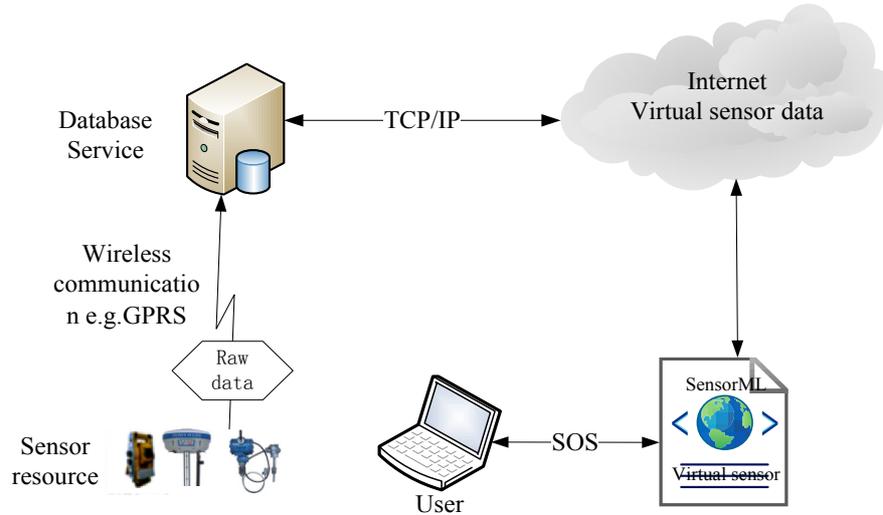


Fig. 4. The data stream from sensor to user(data consumers)

### 4.3 Database Management System

Different sensors not only have different communication protocols but their data formats are not necessarily the same. RS observation data include distance, horizontal angle, and vertical angle from each prism, the GPS receivers get continuous observation data from GPS satellites and those data are big data and the data structure is complex. The database management system does not need to record all the raw data, so we developed middleware to preprocess the raw data. These preprocessed data are called intermediate data and the intermediate data packages are considerably smaller than the raw data. The intermediates data are then stored in the database management system. The smaller data also makes it easier to transmit the data over the Internet. The database management system not only manages intermediate data but it also stores sensor metadata (e.g. the sensor's unique identification code, type, and parameters). The intermediate data and observation timestamp of every sensor were encapsulated into a new virtual sensor [3,22] where no physical sensors were available. These virtual sensors mean that the data center can be considered a virtual sensor network.

#### 4.4 Sensors modeling and registration

A service broker can be deployed on any computer located in the data center that can link to the Internet. In this case, Tomcat service (a web container) is installed on the service broker. Tomcat is a free web application server. Service requesters are the users on the Internet, and they can use web service technology to invoke these web services on the MSMS server. It is necessary to use SensorML to describe the RS and GPS receiver models as two virtual sensors, and then register them on the web service using the RegisterSensor operation of the SOS (describing the sensors makes them known and discoverable). The RegisterSensor operation allows the client to register a new sensor system with the SOS. The Robot model file named “Robot\_model\_Register.xml” is shown in Figure 5. In this file, the “UniqueID” is “MonitorStation\_GEORobotTCA” and it shows the uniqueness of the sensor. There are three “sml:ouput” fields named “EastDif,” “NorthDif,” and “HeightDif.” Those field values are virtual sensor data from the RS via the data management system. Once the producer has registered a sensor with an SOS instance, it can begin publishing observations. The SOS has the responsibility of packaging the observations into offerings and providing them to sensor data consumers (e.g. the application or the users).

When new observational data are sent to the data manager system, an InsertObservation operation is performed. The InsertObservation operation is a request to the SOS to perform the insert operation. The request includes the sensor ID obtained from the RegisterSensor operation. The observation must be encoded in the O&M specification. Figure 6 shows the instructions in a file named “InsertObservaton\_Robot.xml” that performs the show InsertObservaion operation.

Sensor observations are obtained using the GetObservation operation. GetObservation allows the client to filter a large dataset to retrieve only the observations that are of interest. The virtual sensor observation data are sent to the client after consumers submit a GetObservation request to SOS service. The file named “GetObservation\_robot.xml” is shown in Figure 7; it finishes the data request by GetObservation operation.

##### **Robot\_model\_Register.xml**

```

<SensorDescription>
  <sml:SensorML version="1.0.1">
    <sml:identification>
      <sml:IdentifierList>
        <sml:identifier>
          <sml:Term definition="urn:ogc:def:identifier:OGC:uniqueID">
            <sml:value>urn:MonitorStation_GEORobotTCA</sml:value>
          </sml:Term>
        </sml:identifier>
      </sml:IdentifierList>
    </sml:identification>
    <sml:outputs>
      <sml:OutputList>
        <sml:output name="EastingDifference">
          <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:EastDif">

```

```
<gml:metaDataProperty>
  <offering>
    <id>TYUT</id>
    <name>MDMS</name>
  </offering>
</gml:metaDataProperty>
</swe:Quantity>
</sml:output>
<sml:output name=NorthDif">....</sml:output>
<sml:output name=HeightDif">....</sml:output>
<sml:output name=Airpressure">....</sml:output>
<sml:output name=Temperature">....</sml:output>
</sml:OutputList>
</sml:outputs>
</sml:SensorML>
</SensorDescription>
```

Fig. 5. Key code of the RegisterSensor operation

***InsertObservaton\_Robot.xml***

```
<InsertObservation>
  <om:observation>
    <om:procedure xlink:href="urn:MonitorStation_GeORobotTCA"/>
    <om:featureofInterest>
      <om:result>
        <swe:DataArray>
          <swe:elementType name="Components">
            <swe:DataRecord>
              <swe:field name="EastDif">
                <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:EastDif">
                  <swe:uom code="m" />
                </swe:Quantity>
              </swe:field>
              <swe:field name="NorthDif">...</swe:field>
              <swe:field name="HeightDif">...</swe:field>
              <swe:field name="Airpressure">...</swe:field>
              <swe:field name="Temperature">...</swe:field>
            </swe:DataRecord>
          </swe:elementType>
        </swe:DataArray>
      </om:result>
    </om:featureofInterest>
  </om:observation>
</InsertObservation>
```

Fig. 6. Key code of the InsertObservation operation

***GetObservation\_robot.xml***

```
<GetObservation xmlns="http://www.opengis.net/sos/1.0"
  xmlns:ows="http://www.opengis.net/ows/1.0"
```

```
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:om="http://www.opengis.net/om/1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/sos/1.0"
http://schemas.opengis.net/sos/1.0.0/sosGetobservation.xsd"
service="sos" version="1.0.0">
<offering>TYUT</offering>
<procedure>urn:tyut:insitusensor:GPS:AL33</procedure>
<observeProperty>urn:ogc:def:property:OGC:1.0:EastDif</observeProperty>
<observeProperty>urn:ogc:def:property:OGC:1.0:NorthDif</observeProperty>
<observeProperty>urn:ogc:def:property:OGC:1.0:HeightDif</observeProperty>
<responseFormat>text/xml;subtype="om/1.0.0"</responseFormat>
</GetObservation>
```

Fig. 7. Key code of the GetObservation operation

#### 4.5 Client Application

The client application uses the GetObservation operation of the SOS to request near-real-time observation data from the data management database. Because MSMS hosts both a great variety of data and complex data, a geographic information system (GIS) component was used to provide data visualization and graphics operations such as panning and zooming. In view of the types of data collected and the monitoring system targets, we built a model of the dynamic deformation field in the monitoring area that incorporated several space-time data processing methods [23,24]. Figure 8 illustrates overall deformation results of an observation time after analysis. We applied a Gaussian process regression (GPR or kriging) [25] and total least squares [26] to forecast and predict deformation. If the analysis returns a result that is greater than the pre-determine threshold value, an early warning will be issued. For instance, the MSMS will send an alert through the SAS if the coordinates for three points monitored by the SR or the GPS receivers change by more than 20 mm or if an elevation change is greater than 5 cm in a certain time interval. The middle panel in Figure 9 shows the results of Kalman filtering, and the left panel in Figure 9 shows selected reflecting prisms.

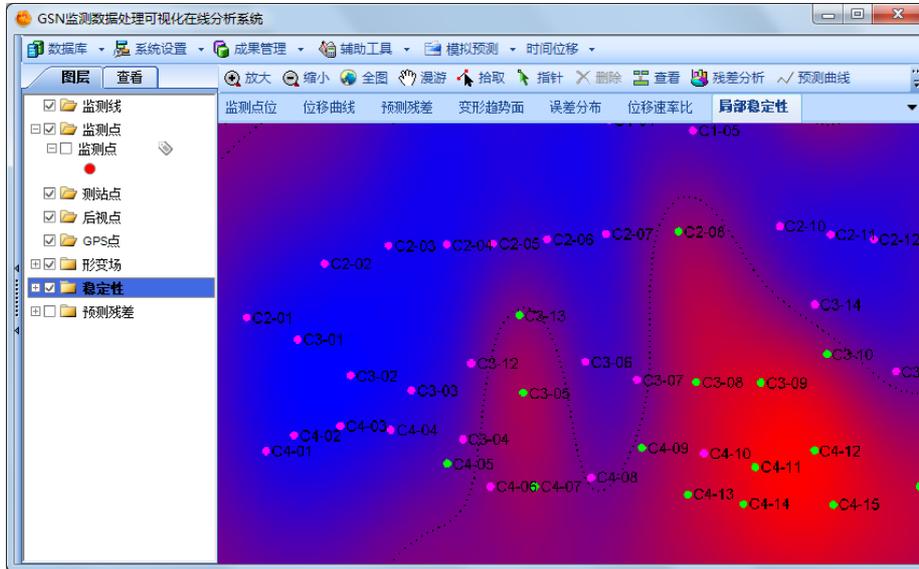


Fig. 8. Example of the results of a deformation observation after near-real-time analysis

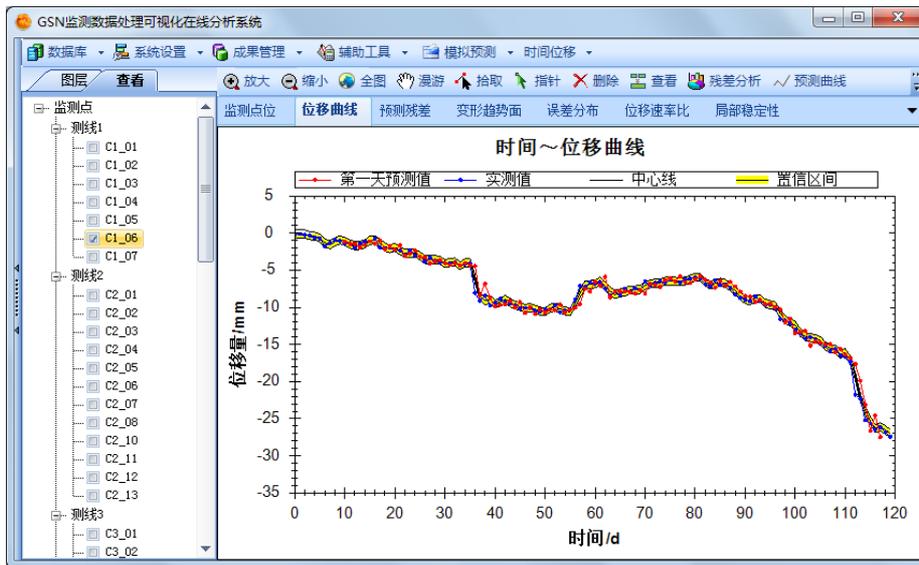


Fig. 9. Example of deformation observations. The middle part shows the results of Kalman filtering

## 5 Conclusions

This paper presents a four layers system architecture for the seamless integration of a multi-sensor remote disaster monitoring system. The system collects data in near-real-time using a geosensor network (GSN) and sensor web technology. In this architecture, every layer plays a different role. The data sensing layer collects the data from the GSN in the monitored area. The data management layer performs the raw data pre-processing and data management. The data service layer is the middleware between the data management layer and the application layer and it deals with virtual sensor data transfers and releases information to the Internet using Sensor Web Enablement. The application layer finishes the data processing and provides data analysis and visualization.

To demonstrate the applicability of this system, we deployed a sensor network, called the Mine Slope Monitoring System (MSMS) on a slope in an opencast coal mine in China. This is a typical application for this type of disaster monitoring system. The deployed system has three types of sensors, a robotic total station surveying instrument, GPS devices, and an air temperature and pressure sensor, plus two assistance systems, a wireless communication system and a solar power supply system. The MSMS uses a Sensor Observation Service (SOS) for sensor data interoperability, data sharing, and the automatic release of disaster early warning notifications.

## 6 Acknowledgment

The work described in this paper was substantially supported by the National Natural Science Foundation of China (grant no. 51504159), the Natural Science Foundation of Shanxi Province, China (grant no. 201701D121014).

## 7 References

- [1] Hu, H. (2016) Online Near Real-time Mine Disaster Monitoring System Based on Wireless Sensor Networks. *International Journal of Online Engineering*, 12: 64–69 <https://doi.org/10.3991/ijoe.v12i03.5450>
- [2] Nittel, S. (2009) A Survey of Geosensor Networks: Advances in Dynamic Environmental Monitoring. *Sensors*, 9: 5664–5678 <https://doi.org/10.3390/s90705664>
- [3] Stasch, C., Foerster, T., Autermann, C., & Pebesma, E. (2012). Spatio-temporal aggregation of European air quality observations in the Sensor Web. *Computers and Geosciences*, 47: 111–118 <https://doi.org/10.1016/j.cageo.2011.11.008>
- [4] Werner-Allen, G., Lorincz, K., Ruiz, M., Marcillo, O., Johnson, J., Lees, J., & Welsh, M. (2006). Deploying a wireless sensor network on an active volcano. *IEEE Internet Comput*, 10: 18–25 <https://doi.org/10.1109/MIC.2006.26>
- [5] Peci, L. M., Berrocoso, M., Páez, R., Fernández-Ros, A., & Gil, .A (2012). Automatic system for monitoring ground deformation on the Deception Island volcano ( Antarctica ). *Computers & Geosciences*, 48: 126–133 <https://doi.org/10.1016/j.cageo.2012.05.004>
- [6] Chiu, J.C., Dow, C.R., Lin, C.M., & Lin, J.H. (2012). A Watershed-Based Debris Flow Early Warning System Using Sensor Web Enabling Techniques in Heterogeneous Envi-

- ronments. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6: 1729–1739 <https://doi.org/10.1109/JSTARS.2011.2181826>
- [7] Worboys, M., & M. Duckham, M. (2006). Monitoring qualitative spatiotemporal change for geosensor networks. *International Journal of Geographical Information Science*, 20: 1097–1108 <https://doi.org/10.1080/13658810600852180>
- [8] Wang, J. (2015). Mine Ground Disaster Monitoring Based on Geosensor Networks. *International Journal of Online Engineering*, 11: 52–56 <https://doi.org/10.3991/ijoe.v11i1.4347>
- [9] Botts, J., Percivall, M., Reed, G., & Davidson, C. (2007). OGC Sensor Web Enablement: Overview and High Level Architecture. OpenGIS White Paper No OGC, 07–165
- [10] Xi, Z., Lu, X., & Li, M. (2012). Metadata Harvesting and Registration in a Geospatial Sensor Web Registry. *Transaction in GIS*, 16: 763–780 <https://doi.org/10.1111/j.1467-9671.2012.01365.x>
- [11] Nittel, S., Labrinidis, A., & Stefanidis, A. (2008). *Geosensor Networks*. Springer LNCS 4540, The Netherlands <https://doi.org/10.1007/978-3-540-79996-2>
- [12] Hsieh, Y., & Hung, Y. (2009). Automation in Construction A scalable IT infrastructure for automated monitoring systems based on the distributed computing technique using simple object access protocol Web-services. *Automation in Construction*, 18: 424–433 <https://doi.org/10.1016/j.autcon.2008.10.010>
- [13] Bröring, A., Echterhoff, J., Jirka, S., Simonis, I., & Everding, T. (2011). New Generation Sensor Web Enablement. *Sensors*, 11: 2652–2699 <https://doi.org/10.3390/s110302652>
- [14] Rio, J., Toma, D.M., Martinez, E., & Jirka, S. (2017). A Sensor Web Architecture for Integrating Smart Oceanographic Sensors into the Semantic Sensor Web. *IEEE Journal of Oceanic Engineering*, 99: 1–13 <https://doi.org/10.1109/JOE.2017.2768178>
- [15] Cox, S. (2007). Observations and Measurements. Open Geospatial Consortium Inc, Available online: <http://www.opengeospatial.org>
- [16] Botts, M., & Robin, A. (2007). OpenGIS Sensor Model Language (SensorML) implementation specification. Open Geospatial Consortium Inc, Tech. Rep, [http://portal.opengeospatial.org/files/%3fartifact\\_id=12606](http://portal.opengeospatial.org/files/%3fartifact_id=12606)
- [17] Jirka, S., Bröring, A., & Stasch, C. (2009). Discovery Mechanisms for the Sensor Web. *Sensors*, 9: 2661–2681 <https://doi.org/10.3390/s90402661>
- [18] Yang, J., Zhang, C., Li, X., Huang, Y., Fu, S., & Acevedo, M.F. (2010). Integration of wireless sensor networks in environmental monitoring cyber infrastructure. *Wireless networks*, 16: 1091–1108 <https://doi.org/10.1007/s11276-009-0190-1>
- [19] Wang, J. & Yang, Y. (2010). Research and Implementation for Sensor Observation Service of Wireless Sensor Network Based on SWE. *Computer & Digital Engineering*, 38: 137–140
- [20] Yang, J., Zhang, C., Li, X., Yan, H., & Fu, S. (2010). Integration of wireless sensor networks in environmental monitoring cyber infrastructure. *Distributed and Parallel Databases*, 16: 1091–1108
- [21] Jiang, R., Song, H., & Zhu, J. (2010). Mine Geological Disaster Supervising and Pre-warning System Based on SensorWeb. *Metal Mine*, 9: 162–165
- [22] Chen, Z., Chen, N., Di, L., Member, S., & Gong, J. (2011). A Flexible Data and Sensor Planning Service for Virtual Sensors Based on Web Service. *IEEE Sensors Journal*, 11: 1429–1439 <https://doi.org/10.1109/JSEN.2010.2095839>
- [23] Wang, J. (2014). Slope deformation analyses with space-time Kriging interpolation method. *Journal of China Coal Society*, 39: 874–879
- [24] Dai, H., Ren, L., Wang, M., & Xue, H. (2011). Water distribution extracted from mining subsidence area using Kriging interpolation algorithm. *Transactions of Nonferrous Metals Society of China*, 21: s723–s726 [https://doi.org/10.1016/S1003-6326\(12\)61669-0](https://doi.org/10.1016/S1003-6326(12)61669-0)

- [25] Wang, J. (2016). Intelligent analysis and prediction of mine slope deformation monitoring data based on gaussian process. PhD thesis, Taiyuan University of Technology.
- [26] Mahboub, V., & Sharifi, M.A. (2013). On weighted total least-squares with linear and quadratic constraints. *Journal of Geodesy*, 87: 279–286 <https://doi.org/10.1007/s00190-012-0598-8>

## 8 Authors

**Jianming Wang** is associate professor of the Department of Surveying and Mapping Science and technology, College of Mining Technology, Taiyuan University of Technology (TYUT), Yingze West St. 79, 030024 Taiyuan, P. R. China. His major is the mine slope deformation monitoring, deformation monitoring data processing and data quality analysis.

**Xiaoqin Yang** is with the College of Mining Technology, Taiyuan University of Technology, Taiyuan, P. R. China.

Article submitted 05 September 2017. Resubmitted 08 January 2018. Final acceptance 05 March 2018. Final version published as submitted by the authors.