Remote Sensing and GIS Applications For Agriculture and Precision Farming

Introducing Software Product Line Development
for Wireless Sensor/Actuator Network Based Agriculture Systems

Mohammad Fajar, Tsuneo Nakanishi, Shigeaki Tagashira, and Akira Fukuda
Department of Advanced Information Technology
Graduate School of Information Science and Electrical Engineering, Kyushu University
744 mootoka, Nishi-ku, Fukuoka 819-0395, JAPAN
{fajar, nakanishi, shigeaki, fukuda}@f.ait.kyushu-u.ac.jp

ABSTRACT

Wireless sensor/actuator networks (WSANs) is one of promising technologies in agriculture domain, since WSANs achieve significant simplification in wiring and reduction of maintenance complexity and costs. So far, researchers have presented various WSAN based applications in agriculture domain such as farm control and monitoring, precision irrigation, traceability management, *etc*. However, these applications are constructed one by one without considering integration and sharing of their software components. But, each farm cultivates different crops under different circumstance. Therefore, in this paper, we perform feature modeling to analyzing commonality and variability among the applications in terms of their features and visualize analyzed commonality and variability in a tree-form diagram. The feature model provides a comprehensive view of the WSAN based agriculture system and helps agriculture domain experts and software engineers communicate intuitively. Moreover, the feature model will be useful for software engineers to pre-design software architecture and reusable components shared by the WSAN based agriculture systems.

Keywords*: software product line, feature modeling, wireless sensor/actuator network, agriculture*

I. INTRODUCTION

In recent years many applications have been proposed for wireless sensor/actuator networks (or WSANs) in agriculture such as environmental monitoring including weather and geo-referenced environmental monitoring, precision agriculture including spatial data collection, precision irrigation and supplying data to farmers, facility automation including greenhouse control and animal-feeding facility, *etc*[14]. These applications require a system that capable to gathering and processing environmental data and then visualizing them to help farmers make various decisions.

In agriculture applications, monitoring and controlling are important activities to support farmer or consumer requirements and to get productivity improvements [12]. To support these activities, WSAN is one of appropriate and promising technologies, since it achieves significant simplification in wiring and reduction of maintenance complexity and costs. Figure 1 shows one of the component in the WSAN, which Kyushu University is operating to gather environmental data of the farm [17].

Each farm cultivates different crops under differentenvironmental, economical, and social circumstances, thus each farm will require different set of agriculture applications. Moreover, each farm will require the set of applications to be executed on its own WSAN platform different from another farm's one
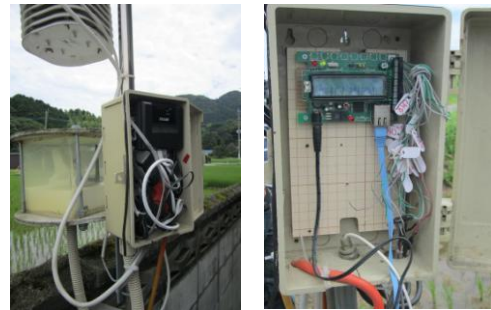

Figure 1: a mote (left: outside view, right: inside view)

. Although researchers have presented various WSAN applications in agriculture domain so far, unfortunately, it seems that these applications tend to be constructed one by one and possibly in a site-specific manner without a ground view of integration and future evolution of various applications. That can causes a considerable amount of efforts to configure the system for each farm and modification of the system to introduce additional features.

The WSAN system for agriculture can be a complicated and large scale software intensive system and diversity of the system mentioned above will be absorbed by software. Software engineering should be applied to reduce design complexity of the WSAN system [4].

In this paper we propose to introduce a paradigm of software product line (or SPL), invented in the field of software engineering, to compose WSAN based systems satisfying different requirements of different farms with shorter term. To reduce configuration and modification efforts for each system, we should construct reusable software components shared by different applications that are deployed in different farms and executed on different WSAN platforms with clearly defined interface. This concept, modularization by reusable software components to compose different systems, has been a key concept of software reuse for a long time. However, bottom-up reuse approach without a ground vision of the system family has often resulted in components that are hardly reusable in other systems, causes frequent modification, and requires costs for their maintenance. SPL is different from previous reuse methodologies in terms that it aims composition of systems in the pre-scoped family with pre-planned reusable components based on the ground vision of the system family.

The organization of this paper is as follows: Section II discusses WSAN and SPL concepts, especially feature modeling and its example. In Section III we talk about the related works. We present WSAN architecture, separation between logical and physical level, logical and physical level feature modeling and their relationship in Section IV. Finally, our concluding remark and future works are presented in Section V.

II. KEY TECHNOLOGIES AND CONCEPTS

A. Wireless Sensor/Actuator Network

The wireless sensor/actuator network (or WSAN) is a large-scale *ad hoc*, multi-hop, un-partitioned network of largely homogeneous, tiny, resource-constrained, mostly immobile sensor nodes that would be randomly deployed in the area of interest [10]. For example, Hydra Project [5] uses battery-powered motes with a limited amount of resources. At the same time, in agriculture domain, there are approaches to use motes with rich facilities such as CCD camera, wire-less LAN, *etc.* like Field Server Project [16]. The mote itself can have variability, thus we should select the mote depending on applications, requirements, and affordable investment of each farm.

Figure 3 in Section IV shows a typical WSAN based system for agriculture, which consists of sensor nodes (or motes), gateways, a base station, repeaters and a PC network.

B. Software Product Line (SPL)

SPL is a paradigm of software reuse for the system family. In SPL, the system family is defined as a set of software intensive systems that share a common, managed feature set satisfying a particular market segment's specific needs or mission and that are developed from a common set of core assets in a prescribed way [13]. The main motivations for SPL are: reduction of the development cost, improvement of software quality, and reduction of time to market, this is because many artifacts can be reused for each new product. Software reuse has positive impacts on software quality, software cost, productivity, performance, reliability and interoperability.

Software artifacts have often been reused in an *ad hoc* manner. To develop a new system, engineers extend and modify the existing system that they developed before. This clone-and-own approach lacks a ground vision of the system family. Everything is developed in a single product focus where resources will not be spent on developing variability mechanisms enabling reuse and sharing of artifacts across multiple systems. That implies a new maintenance process as well as additional maintenance costs for the new system. Moreover, the software architecture will get more complicated and error-prone as additional features are introduced in the system family. A corrupting software architecture provides a limited restriction to the structure and the behavior of the software components. That will degrade reusability of existing software components.

Otherwise, in the SPL paradigm, we define the software architecture with variability mechanisms shared in the system family according to the ground current and future vision of the system family. We develop and maintain software components applied to the architecture to compose different systems in the family. The set of the components are maintained as a core asset, a set of artifacts stored in a shared repository and reused in the multiple members of the system family, with variability mechanisms. The core asset includes not only implementation artifacts such as software components but also requirement, analysis, design, and test artifacts. This architecture centric component reuse contributes to avoid corruption of the software architecture due to changeable and additional requirements by different farms and keep reusability of the software component higher.

Figure 2 shows a simple illustration of the SPL approach. The software architecture is shared by all the members of the system family. Common functionalities that all the members of the system family must equip are realized by common components applied to the software architecture. Variable functionalities that each member of the system family may or may not equip are realized by combination of the different components attached to the variation points of the software architecture. Components responsible for these common and variable functionalities are stored in the core asset for maintenance.
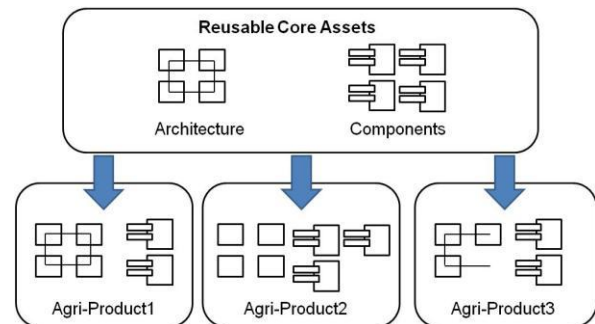

Figure 2: Architecture Centric Reuse in SPL

C. Feature Modeling

In SPL, commonality is a quality or function that all the members in a system family share. As a consequence, commonalities are the elements with the highest reuse potential because an implementation of a commonality is used in all family members. In contrast, a variability is the ability to change or customize a system[7]. Variability always represents a variable aspect from the viewpoint of the requirements (*i.e.* they abstract from variable functionality or a variable quality of the system) [11]. Commonality and variability modeling is an important activity in SPL, especially in requirements analysis of the system.

Feature modeling [8] is a well-known and widely used technique to represent commonality and variability of the system family. In feature modeling, commonality and variability among members of the system family are described in terms of the *features* that each member of the system family equips. The feature is a prominent or distinctive concepts or characteristics that are visible to various stakeholders [9]. For example, we can find various features that WSAN based agriculture systems have.

As a result of feature modeling, we will have a feature model represented as a tree form diagram, a feature diagram, that describes features with annotation of variability classification and relationships among the features. Figure 5 shows example feature models for the WSAN based agriculture system family and the WSAN mote family.

The node of the feature diagram represents a feature. In Figure 5, there are features found in the WSAN based agriculture system such as *Farm Maintenance*, *Crop Maintenance*, *Decision Making Assistance*, *etc.*

Relationships between features are represented as tree edges in the feature diagram. There are three kinds of relationships between features: consists-of relationship, generalization/specialization relationship, and implemented-by relationship. A consists-of relationship represents that the feature corresponding to the parent node consists of one or more sub features corresponding to the children nodes. A generalization/specialization relationship represents that the feature corresponding to the parent node is generalized from one or more features corresponding to the children nodes. An implemented-by relationship represents that the feature corresponding to the parent node is implemented by the one or more features corresponding to the children nodes. For example, according to Figure 5, we know that i) *WSAN Agriculture Application Family* consists of *Farm Maintenance*, *Crop Maintenance*, *Decision Making Assistance etc*. and *Farm Maintenance* consists of *Farm Security etc*., ii) *Battery* is generalized from *Rechargeable Battery* and *Non-Rechargable Battery*, and iii) *Automatic Locating* is implemented by *GPS*.

In the original feature diagram[8], there are three kinds of variability classification of the feature: mandatory, optional, and alternative. The mandatory feature must be equipped by all the members of the system family. The optional feature may and may not be equipped by each member of the system family. The alternative features in a group are features that should be exclusively selected by each member of the system family. These variability classifications are represented by decoration to the nodes and the edges in the feature diagram.

A node with a white circle represents an optional feature. A set of alternative features are represented by an arc that bundles the edges between alternative features and their common parent feature. For example, according to Figure 5, we know that i) *Local* Environmental *Sensing* is a mandatory feature of the WSAN mote family, ii) *Image Capturing* is an optional feature of the WSAN mote family, iii) *Manual Locating* and *Automatic Locating* are a set of alternative features of the WSAN mote family.

Note that feature relationships represented by tree edges imply dependency between features, that is, the feature corresponding to the child node of the tree edge cannot be equipped unless the feature corresponding to the parent node of the same tree edge is equipped. However, there can be dependency between features neither in consists-of, generalization/specialization, nor implemented-by relationships represented by the tree edge. To represent such a dependency, we can use a dependency edge across the tree edge in the feature diagram. The dependency edge represents that the feature corresponding to its source requires the feature corresponding to its sink to be equipped. For example, according to Figure 5, we know that suggestion of *Fertilization* requires Field Environment Sensing of Soil *Fertility* for its realization.

III. RELATED WORK

As a first stage in our works, we reference some results of research works in the WSAN implemented in the agriculture domain. The results are used to capture features widely found in existing WSAN based agriculture systems.

Some works describe construction of WSAN based systems for agriculture [2,3,5,6,12,16,17,19,20] and some other works describe experimental results on WSAN based systems [1,15]. The WSAN technology is applied not only for crop fields but also for pastures. The work in[18] employs the WSAN technology for pasture assesment.

There is also a paper describing recent development and future perspective of the WSAN technology in agriculture and rural area [14].

IV. FEATURE MODELING

In this section we perform feature modeling to sketch a ground vision of the WSAN based agriculture system family. The feature model, a result of feature modeling, will contribute to give an overview of WSAN based agriculture applications and WSAN platforms. Moreover, the feature model helps us pre-design the software architecture of the system family because the feature model layouts features depending on their abstraction level in a tree form.

A. WSAN Architecture

To perform feature modeling, we have to assume a WSAN architecture on where we perform various agriculture applications. The architecture should be enough typical to guarantee the system family can include the members deployed in a huge variety of our assuming farms. Figure 3 is the WSAN architecture that we assume for feature modeling in this paper.

The WSAN in agriculture is a collection of motes, gateways, repeaters, a base station, and a PC network. The network may provide an internet access.

The mote with sensors, which is placed in the farm, is responsible forsensing an environmental data and transferring the data to other motes or gateways with a wireless network. The mote may perform simple dataprocessing. The gateway is responsible for transferring data between the motes and the farm office. The repeater just resends the data it receives. The base station is a port of the farm office network to the mote network in the farm. The PC provides sophisticated processing services for the data collected from the WSN network and interface services for the farmer.
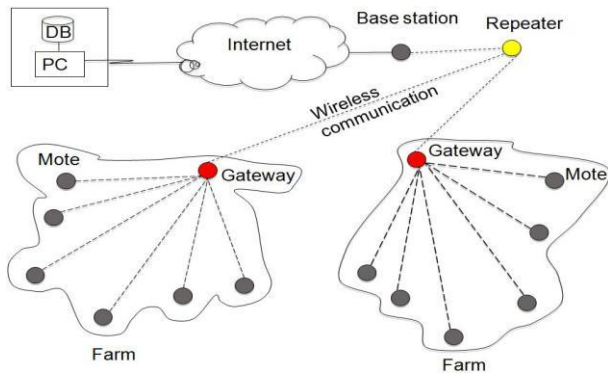

Figure 3: WSAN Architecture

B. Separation of Logical and Physical Layers

We separate modeling works, including feature modeling, of the WSAN based agriculture system family into two layers: logical and physical layers.

The feature model in the logical layer represents features and relationships among them for applications executed in the WSAN architecture. Each feature of the applications shown in the logical layer feature model can be realized as a collaborative work of the components in the WSAN architecture.

The feature model in the physical layer represents features and relationships among them for each component of the WSAN architecture such as mote, repeater, gateway, base station *etc.* because it can have variants and form a family.

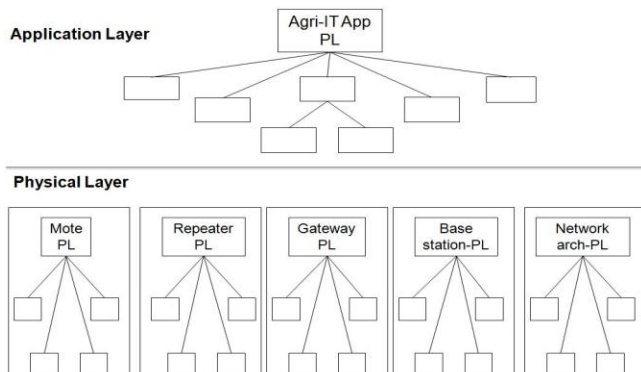Figure 4 shows the concept of separation of logical and physical layers.

Figure 4: Separation of Logical/Physical Layers

C. Logical Level Feature Modeling

To perform logical level feature modeling, we extract some distinct services as features from the existing WSAN systems appeared in published works. Moreover, we add our original services according to our personal surveys and interviews to farmers. The feature model in the upper side of Figure 5 shows a partial result of logical level feature modeling.

Looking existing WSAN systems for agriculture, most of the systems collect environmental data such as air temperature, air humidity, soil temperature, solar radiation *etc.* at the place at where the mote is located. Therefore, we can find these features easily. Moreover, we can group these features depending on the object to be measured by the mote, namely air, soil, water, *etc.* The logical level feature modeling in Figure 5 shows this hierarchical structure of *Field Environmental Sensing*.

Field environmental sensing itself is not a main purpose of the WSAN based agriculture system family. We have to think the reasons why field environmental sensing is performed. Thinking the reasons, the services provided by the WSAN based agriculture system emerge. In this way, we can find service features which bring valuable functionalities to the stakeholders of the system such as *Decision Making Assistance* including *Suggestion* of various agricultural activities, *Visualization* of location dependent sensed data, *etc.*

After organizing hierarchical structure of the features by tree edges, we check dependency among features. The dependency which cannot be represented by the tree edge is represented by a required edge.

D. Physical Level Feature Modeling

The features in the logical level feature model are realized by the physical components of the WSAN architecture including motes, gateways, *etc.* These physical components of the WSAN architecture have variability, too. Therefore, we have to perform feature modeling for each physical component independently from the logical level variability, namely variability of the WSAN applications.

The feature model in the lower side of Figure 5 shows a partial result of physical level feature modeling. The way of thinking of physical level feature modeling is similar for that of the logical level feature modeling.

E. Relationship between Logical and Physical Layers

The feature in the logical level feature model possibly requires features in the physical level feature model for its realization. This kind of dependency from the feature in the logical feature model to the feature in the physical feature model should be described by a required dependency edge.

For example in Figure 5, *Field Environmental Sensing* in the logical level feature model requires *Local Environmental Sensing* in the physical level feature model to provide its services. Here, *Field Environmental Sensing* is a service provided by the entire system to gather data sensed by the motes deployed in the field. On the other hand, *Local Environmental Sensing* is a service provided by each mote to sense environmental data.
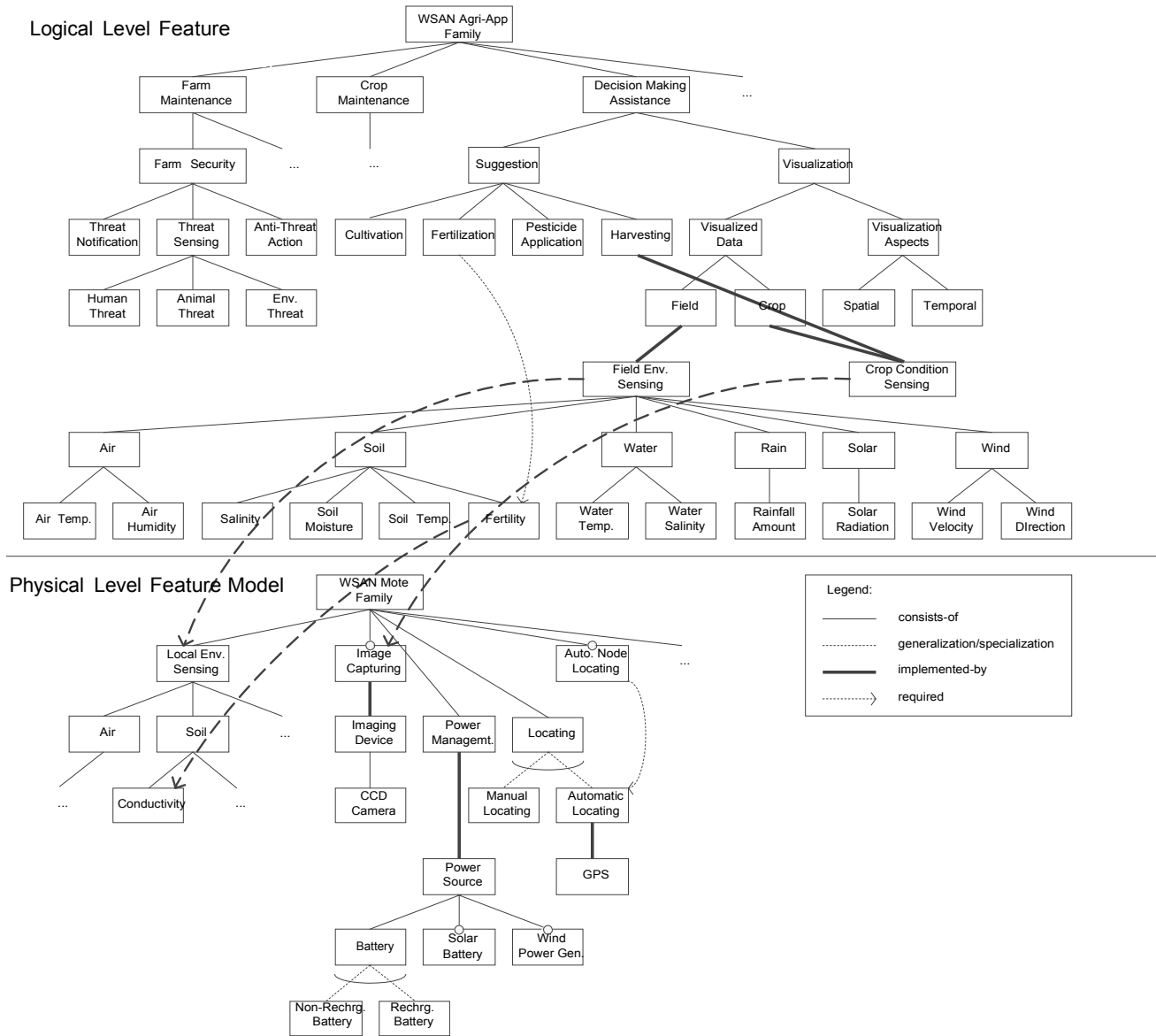
Figure 5: Feature Model of the WSAN Based Agriculture System Family

To provide *Field Environmental Sensing*, the system will use *Local Environmental Sensing* provided by the motes in the WSAN. This is a reason for the required dependency from *Field Environmental Sensing* to *Local Environmental Sensing*.

V. CONCLUSIONS

We proposed to introduce a paradigm of software product line (or SPL) in WSAN based agriculture systems which has been developed one by one for a certain purpose so far. To enable construction of the WSAN based agriculture system family, we present a framework of the feature model. In the framework, we perform logical level feature modeling for commonality and variability analysis of agriculture applications executed on the entire WSAN architecture as well as physical level feature modeling for

commonality and variability analysis of the component of the WSAN architecture. We also showed the example feature models of the WSAN based agriculture system family. The feature model will contribute to show a comprehensive view of current and future WSAN based agriculture systems and help agriculture domain experts and software engineers communicate intuitively. The feature model will be useful for pre-design of the software and reusable components shared by the WSAN based agriculture systems.

For future works, we must improve the feature model through discussion with agriculture domain experts, define the scope of the WSAN based agriculture system family to be implemented, derive the software architecture for the system family and construct the core asset for quick derivation of the system to be deployed in different farms.

REFERENCES

[1]   A. Baggio, "Wireless Sensor Networks in Precision Agriculture," *Proc. ACM Workshop Real-World Wireless Sensor Networks*, 2005.

[2]   A. Hasimoto *et al*, "Remote Monitoring of Color of Agriculture Products in the Field Using a Digital Camera and the Field Server," *Proceedings of the 4ᵗʰ World Congress Computers in Agriculture and Natural Resources*, pp.66-71, 2006.

[3]   F. J. Pierce and T. V. Elliott, "Regional and on-farm wireless sensor networks for agricultural systems in Eastern Washington," Computers and Electronics in Agriculture, Vol.61, No.1, pp.32-43, 2008.

[4]   Ian Sommerville,*Software Engineering*, 7ᵗʰ edition, Addison Wesley, 2004.

[5]   J.A. Lopez Riquelme, F. Soto, J. Suardiaz, P. Sanchez, A. Iborra, and J.A. Vera, "Wireless Sensor Networks for Precision Horticulture in Southern Spain," Computers and Electronics in Agriculture, Volume
68, Issue 1, pp.25-35, August 2009.

[6]   J.Burrel, T.Brooke, and R.Beckwith, "Vineyard Computing: Sensor Networks in Agriculture Production," IEEE Pervasive, pp.10-17, 2004.

[7]   J.V. Gurp, J.Bosch, and M.Svahnberg, " On the Notion of Variability in Software Product Lines," *Proc. of the Working IFIP Conference on Software Architecture (WICSA 2001)*, p.45, August 2001.

[8]   K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Novak, and A. S. Peterson, "Feature-Oriented Domain Analysis (FODA) Feasibility Study," Technical Report CMU/SEI-90-TR-21, SEI/CMU, Nov. 1990.

[9]   K.Lee, K.C. Kang, and J.Lee, " Concepts and Guidelines of Feature Modeling for Product Line Software Engineering," Springer-Verlag Berlin Heidelberg, pp.62-77, 2002

[10] K.Romer and F.Mattern,"The Design Space of Wireless Sensor Networks," IEEE Wireless Communications, pp.54-61, December 2004.

[11] L.Geyer and M.Becker, "On the Influence of Variabilities on the Application Engineering Process of a Product Family," *Proc of the Software Product Lines Second International Conference, SPLC 2*, san diego,*CA,USA, Springer*, pp.1-14, August 2002.

[12] L.Gonda and C.E. Cugnasca,"A Proposal of Greenhouse control using wireless sensor networks," *Proceedings of the 4ᵗʰ world congress Computers in Agriculture and Natural Resources*, pp.229-
233, 2006

[13] L.M. Northtrop, "SEI's Software Product Line Tenets," IEEE Software, pp.32-40, 2002.

[14]   N.Wang, N.Zhang, and M.Wang, "Wireless Sensors in Agriculture and Food Industry: Recent Development and Future Perspective," Computers and Electronics in agriculture, vol 50, pp 1-14, January
2006.

[15] R. Beckwith, D. Teibel, and P. Bowen, "Report from the Field: Result from an agriculture wireless sensor network," *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*,pp.471-478, 2004.

[16] T. Fukatsu, M. Hirafuji, T.Kiura, "A Distributed Agent System for Managing a Web-based Sensor Network with Field Servers," *Proc. of 4th World Congress on Computers in Agriculture (WCCA)*, pp.223-228. 2006.

[17] T.Okayasu,N. Yamabe, A. Marui, T.Miyazaki, M.Mitsuoka, and E.Inoue, "Development of Field Monitoring and Work Recording System in Agriculture," ISMAB2010, CD-ROM, 2010.

[18] T.Wark *et al*., "Transforming Agriculture through Pervasive Wireless Sensor Networks," IEEE Pervasive, pp.50-57, 2007.

[19] W. Zhang, G. Kantor, and S. Singh, "Integrated Wireless Sensor/Actuator Networks in an Agricultural Application," *Proc. 2nd ACM Int. Conf. Embedded Networked Sensor Systems (SENSYS04)*,
p.317, 2004.

[20] Y.Wang and G.Weng, "The Monitoring Population Density of Pests Based on Edge-Enhancing Diffusion Filtering and Image Processing," IFIP International Federation for Information Processing, Volume 259; Computer and Computing Technologies in Agriculture, Vol. 2; Daoliang Li; (Boston: Springer), pp. 899–907, 2008