



HAL
open science

Réseaux de capteurs sans fil pour l'observation du climat et de la biologie dans une région tropicale d'agriculture intensive: méthodes, outils et applications pour le cas du Delta du Mékong, Vietnam

Bao Hoai Lam

► **To cite this version:**

Bao Hoai Lam. Réseaux de capteurs sans fil pour l'observation du climat et de la biologie dans une région tropicale d'agriculture intensive: méthodes, outils et applications pour le cas du Delta du Mékong, Vietnam. Other [cs.OH]. Université de Bretagne occidentale - Brest, 2018. English. NNT : 2018BRES0006 . tel-01871886

HAL Id: tel-01871886

<https://theses.hal.science/tel-01871886>

Submitted on 11 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



université de bretagne
occidentale

UNIVERSITE
BRETAGNE
LOIRE

THÈSE / UNIVERSITÉ DE BRETAGNE OCCIDENTALE

sous le sceau de l'Université Bretagne Loire

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITÉ DE BRETAGNE OCCIDENTALE

Mention : Science et Technologie de l'Information et de la Communication
École Doctorale Mathématiques-STIC (MathSTIC) ED 601

présentée par

Bao Hoai LAM

Lab-STICC UMR CNRS 6285

Université de Bretagne Occidentale, France

Can Tho University, Vietnam

**Réseaux de capteurs sans fil pour
l'observation du climat et de la
biologie dans une région tropicale
d'agriculture intensive.
Méthodes, outils et applications
pour le cas du Delta du Mékong,
Vietnam**

Thèse soutenue le 26 Janvier 2018 devant le jury
composé de:

Monsieur Christophe GUINET

Directeur de recherche, CNRS/CEBC Chizé / *Rapporteur*

Monsieur Congduc PHAM

Professeur, Université de Pau et Pays de l'Adour / *Rapporteur*

Monsieur Hiep Xuan HUYNH

Enseignant-Chercheur HDR, Université de Can Tho / *Examinateur*

Madame Simona NICULESCU

Maître de conférences - HDR, Université de Bretagne Occidentale /
Examinatrice

Monsieur Vincent RODIN

Professeur, Université de Bretagne Occidentale / *Examinateur*

Monsieur Bernard POTTIER

Professeur, Université de Bretagne Occidentale / *Directeur de thèse*

Monsieur Jean-François DORVILLE

Enseignant-Chercheur Dr, Université de Antilles / *Invité*



Université
de Bretagne
Occidentale



THÈSE / UNIVERSITÉ DE BRETAGNE OCCIDENTALE Présentée par

Sous le sceau de l'Université Européenne de Bretagne

Bao Hoai LAM

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE BRETAGNE OCCIDENTALE

Lab-STICC UMR CNRS 6285

Mention: Science et Technologie de l'Information et de la Communication

Université de Bretagne Occidentale, France

École Doctorale Mathématiques-STIC (MathSTIC)

Can Tho University, Vietnam

ED 601

**Réseaux de capteurs sans fil pour
l'observation du climat et de la
biologie dans une région tropicale
d'agriculture intensive. Méthodes,
outils et applications pour le cas du
Delta du Mékong, Vietnam**

**Thèse soutenue le 26 Janvier 2018 devant le jury
composé de:**

Monsieur Christophe GUINET

Directeur de recherche, CNRS/CEBC Chizé / *Rapporteur*

Monsieur Congduc PHAM

Professeur, Université de Pau et Pays de l'Adour / *Rapporteur*

Monsieur Hiep Xuan HUYNH

Enseignant-Chercheur HDR, Université de Can Tho / *Examinateur*

Madame Simona NICULESCU

Maître de conférences - HDR, Université de Bretagne Occidentale /
Examinatrice

Monsieur Vincent RODIN

Professeur, Université de Bretagne Occidentale / *Examinateur*

Monsieur Bernard POTTIER

Professeur, Université de Bretagne Occidentale / *Directeur de thèse*

Monsieur Jean-François DORVILLE

Enseignant-Chercheur Dr, Université de Antilles / *Invité*

Abstract

Climate changes bring problems related to nature evolutions. Global warming has an impact on sea level, weather patterns, and wild life. A number of national and international organizations are developing research programs in these directions, including threats on cultures and insect proliferation. Monitoring these phenomena, observing consequences, elaborating counteracted strategies are critical for the economy and society.

The initial motivation of this work was the understanding of change impacts in the Mekong Delta region. From there, automatic observation tools were designed with a real time information system able to integrate environmental measures, then to support knowledge production.

Tracking environment evolutions is distributed sensing, which can be the association of efficient sensors and radio communications, operated under the control of an information system. Sensing insects is very complex due to their diversity and dispersion. However, this is feasible in the case of intensive agricultural production as it is the case of rice, having a small number of pests. An automatic vision observatory is proposed to observe the main threats for the rice, as an evolution of manual *light traps*. Radio communication weaves these observatories into a network with connection to databases storing measures and possible counteractions. An example observatory has a fisheye camera and insect counting algorithms for the BPH practical case in Vietnam.

By considering the observation system as an input for an abstract machine, and considering decision and actions taken as a possible control on the environment, we obtain a framework for knowledge elaboration that can be useful in lots of other situations.

The detail of chapters is as follows: global warming context, automatic light traps design, "*cyber-physical*" environment machine concept, and its supporting information system. A study of insect invasion in Mekong Delta, Vietnam is used as an illustration. It advocates feasibility by a starting implementation.

Keyword: climate change, vision system, light trap, fisheye lens vision, insect counting, Brown Planthopper, environment machine, cellular automata, synchronous network, geographic information system

Résumé

Les changements climatiques ont des impacts considérables sur le temps, les océans et les rivages, la vie sauvage. Ils amènent des problèmes désormais considérés comme majeurs par les gouvernements et organisations internationales. La région du Delta du Mékong subit actuellement des évolutions hydrologiques et biologiques contre lesquelles des efforts d'observation et de compensation sont entrepris.

Ces efforts ont fourni un cadre à cette thèse, qui propose de procéder en boucle fermée de l'observation d'insectes ravageurs, avec des centaines de capteurs en réseau ("light traps"), au système d'information, et enfin à des décisions de lutte, manuelles ou automatiques. Le point d'appui pratique est la conception d'un système de comptage d'insectes proliférant dans les cultures de riz (BPH). L'abstraction que nous développons est celle d'une machine environnementale de grande taille, distribuée, qui capte et synthétise l'information, élabore des connaissances, et prend des décisions. Autour de cette abstraction, nous avons élaboré un système de vision "fisheye" effectuant le comptage des insectes.

Nous proposons un système d'information géographique directement connecté au réseau de capteurs.

Le couplage direct, "cyber-physique", entre les systèmes d'information et l'observation de l'environnement à échelle régionale est une nouveauté transposable, qui permet de comprendre et contrôler quantité d'évolutions.

Ces travaux sont partie intégrante du projet STIC-Asie SAMES financé par le Ministère des Affaires Etrangères (UBO/LabSTICC et al.). Le gouvernement du Vietnam a financé la thèse en liaison avec l'Université de Cantho (CTU).

Mots clés : changement climatique, système de vision, piège lumineux, caméra fisheye, classification comptage d'insectes, Cicadelle brune, machine environnementale, automates cellulaires, réseau synchrone, système d'information géographique

Acknowledgement

At the end of this writing, I am aware that research is far from a solitary work. Indeed, I could never have ever finished the doctoral study without the support of a large number of people who are generous, committed and interested in the scientific contents.

First, I would like to express my deep appreciation to Mr Bernard POTTIER, Professor at Université de Bretagne Occidentale (UBO), France. He welcomed me with open arms to LabSTICC and agreed to supervise this thesis. I don't know how much I appreciate his great availability, patience, support, listening and understanding throughout this research work.

Mr Hiep Xuan HUYNH, Associate Professor at Can Tho University, Vietnam, was a key person for introducing me to UBO, directing and explaining the insect case which are critical in the Mekong Delta. Special thanks go to him and his team (Ky Minh NGUYEN, Anh Tuan TRAN) for their recommendation and supporting me in the thesis work.

I would like to state my gratitude to Mr. Germán FABREGAT, Associate Professor at High Performance Computing and Architectures (HPCA) group, Universitat Jaume I (UJI), Spain for his advice and support in using fisheye lens camera to classify/count insects. Warm thanks are also given to people in the HPCA group for their welcoming and giving advice for me during the 3 months stay in UJI. I can't forget to thank UBO which gave me a chance to exchange study with a '*mobilité internationale bourse*' to go to UJI from September 2016 - December 2016.

I am really grateful to Dr. Jean-François DORVILLE from Université des Antilles, France, for his reading and review of my thesis. He gave me comment and advice to improve my writing skills. Besides, I would like to thank Mr. Vincent RODIN, Professor at UBO, due to his review of my document.

My sincere thanks also go to members of the jury:

- Mr Christophe GUINET, *Directeur de Recherche* of CNRS/CEBC Chizé, reviewer.
- Mr Congduc PHAM, *Professeur* at *Université de Pau et Pays de l'Adour*, reviewer.
- Mrs Simona NICULESCU, *Maître de conférences - HDR* at *UBO*, examiner.

Please let me send my thanks to those who supported me and shared unforgettable moments with me during my stay in Brest.

These appreciations would be incomplete if I forgot my wife, Duyen Hong NGUYEN who accepted me to study abroad, took care of my son - Huy Bao LAM, supported and encouraged me during my study in France.

The work was in the scope of the "OBSNET - Optimizing the Brown Planthopper surveillance network using automatic light traps in the Mekong Delta region" project of the Ministry of Education and Training, Vietnam. I was also funded by the program 911 of that Ministry. Please let me express my deep gratitude to those who are responsible for these programs.

The cooperation between Can Tho University and UBO was part of the STICAsie program SAMES, Ministry of Foreign Affairs, France, strengthening a lot this work.

1

Introduction

1.1 Climate change

1.1.1 Impacts

Climate change is an alteration in typical or average weather of region, city or country during a long period of time, e.g decades, century or millennium. Human activities may promote the ongoing process, often referred to global warming [37]. This Global Warming can be caused by natural processes and/or human activities. However, the meaning of that term, which is commonly used nowadays, is mainly considered as produced by human activities. Actually, those two terms have different senses since the last is about the rising trend in average temperature of earth while the first one is a wider concept which refers to long-term changes in the climate such as temperature, rainfall, sea level rise as well as other effects on nature and human being.

Effects of climate change vary from average temperature and sea level rise to increase

of frequency of natural disasters and extreme weather.

The Asia-Pacific region is considered as one of the most vulnerable and affected regions. Many kinds of natural disasters, climatic events and extreme weather happen with increase of scale, frequency and intensity. The 2015 report of United Nations Economic and Social Commission for Asia and the Pacific [178] shows that, out of a total of 90 storms and 344 disaster events in the world, 47% of those occur in Asia-Pacific countries and territories.

The world wide average temperature has increased rapidly since the beginning of the Industrial Revolution. In the fourth assessment report taken by Intergovernmental Panel on Climate Change (IPCC) [35], from 1906-2005, average global temperature had an increase of 0.74°. In the last 50 years, average temperature rose twice in comparison with previous years. The 1990s was the hottest decade since 1861, even in the last 1000 years in the Northern Hemisphere.

Average sea level increased approximately of 1.8 mm/year in the period 1961-2003 [35]. It is due to thermal expansion of water and melting of continental ice at poles and on high peak mountains.

Extreme weather conditions have been recorded since 1950:

- In most continents, number of cold days and nights have declined while number of warm days and nights have been increased.
- There are evidences of increasing of sunny days in Asia, Africa and South America and growth of global number of rainy days recorded.
- Large tidal waves tend to happen more often due to sea level rise in the second part of the 20th century.

Climate change is a disadvantage for countries for which agriculture is the main sector of economy (most of them are located in Africa and Asia) [42]. Indeed, high temperatures, erratic rainfall directly affect productivity/yield of crops and threaten their food security. For example, in Nigeria, due to climate change, productivity of plants is expected to decrease from 10% to 20%, even up 50% in 2050 [87]. At the same time, the agricultural activities have remarkably increased climate change using old methods producing significantly amounts of greenhouse gases [116].

1.1.2 Global strategies

There are two strategies to confront with climate change: mitigation and adaption.

- Mitigation prevents global warming by reducing greenhouse gas emissions. In fact, it addresses root causes and allows limitation magnitude or rate of long-term changes [52].
- Adaption is about anticipating the adverse effects to prevent or reduce their damages. It includes human activities to adapt and enhance resilience to impact so people can exploit its favorable aspects.

Researches in this field can be divided in three major groups [138]:

- Nature, causes, and physical mechanisms. The mission is to assess, answer questions about evidence of modern climate change, demonstrate its causes, show ability of climate models to simulate past and present, rationality of greenhouse gas emission scenarios as well as produce projections of future climate.
- Assess impacts, vulnerability to climate change and adaptation. The mission is to assess impact, vulnerability, resilience and strategies, and action plans for adaptation. The main issue is that climate change can bring benefits to a number of subjects, sectors and areas, but this assessment here is mainly emphasized in terms of its negative impact.
- Solutions, strategies and action plans to mitigate. The mission is to find solutions for mitigation. Mitigation term is understood in the goal to reduce green house gas emissions to keep earth's climate stable within a certain limits. It is also motivated by development of clean technologies, the production and use of green energies.

1.1.3 Past and current international efforts

Climate change has received attention of most countries and global organizations in the world ¹. United Nations and Food & Agriculture Organization of United Nations put a lot of effort into realization, including:

United Nations

The United Nations support the fight against climate change via convention and agreement, as follow:

¹https://en.wikipedia.org/wiki/World_Climate_Conference

- United Nations Framework Convention on Climate Change. It is an environmental international agreement negotiated at the United Nations Conference on Environment and Development (UNCED), commonly known as the Earth Summit in Rio de Janeiro, Brazil from 3 - 14/06/1992. Its goal is that *the Conference of the Parties may adopt is to achieve, in accordance with the relevant provisions of the Convention, stabilization of greenhouse gas concentrations in the atmosphere at a level that would prevent dangerous anthropogenic interference with the climate system* (Article 2 in [189]). As of September 2017, UNFCCC has 198 parties and observers ².
- Kyoto Protocol [190]. It extends the 1992 United Nations Framework Convention on Climate Change of the United Nations with the goal of reducing greenhouse gas emissions. It was adopted in Kyoto, Japan, on 11/12/1997, and entered into force on 16/02/2005. An important content of Kyoto Protocol is that industrialized countries reduce *their overall emissions of greenhouse gases by at least 5% below 1990 levels in the commitment period 2008 to 2012*.
- Paris Agreement [191]. It is an agreement within the United Nations Conference on Climate Change 2015 under the United Nations Framework Convention on Climate Change (UNFCCC) that governs carbon dioxide emission reductions from 2020. This agreement was negotiated at the 21st Conference of the Parties to the UNFCCC and adopted December 12, 2015. Some main contents relates to *holding the increase in the global average temperature to well below 2°C and pursuing efforts to limit the temperature increase to 1.5°C*. Moreover, 18 industrialized countries state that by 2020, they provide \$100 billion a year to developing countries from a wide variety of sources, for meaningful mitigation actions ³.

Food and Agriculture Organization of the United Nations - FAO

To FAO, climate change is put as a top line priority by following main activities [56]:

- Agriculture prioritizes adaptation since changing of climatic conditions, which negatively affect rural livelihoods as well as food security. Between 2003 and 2013, natural hazards and disasters in developing countries have cost \$494 billion

²http://unfccc.int/parties_and_observers/parties/items/2352.php

³<http://newsroom.unfccc.int/financial-flows/18-industrial-states-release-climate-finance-statement/>

and affected 1.9 billion people [55]. However, there is no report about damage in agriculture sector, although the sector is important to economy. There is a clear gap in terms of disasters damage on agriculture and livelihoods as well as food security of populations. FAO makes effort to fill this gap.

- Supporting countries to deal with impacts. Agriculture sector is very sensitive to climate change. From assessment of post-disaster needs 2003-2013 data [53], agricultural activities, including crops, livestock, fisheries and forestry, absorb 22% of the total economic impact caused by natural hazards. Those activities, combining with global food production and livelihoods, allow integrating agriculture and adaptation efforts.
- Data, methods and tools. They are created to inventory and measure of emissions as well as for assess risks and vulnerabilities. For example, Locust Watch is a system designed to monitor desert locusts, weather and ecological conditions and it is used to provide global early warning system for transboundary plant pests and diseases ⁴. Another example is databases on green house gases emissions from agriculture and relating sectors. They are maintained to provide a coherent and internationally neutral data platform ⁵.
- Actions on ground to address adaptation and mitigation in agricultural sectors. For example, UN-REDD program ⁶ is an effort of FAO with United Nations Development Programme (UNDP) and United Nations Environment Programme (UNEP) to create financial value from carbon stored in forests, for reducing emissions caused deforestation and forest degradation (REDD) in 64 partner countries across Africa, Asia-Pacific, Latin America and the Caribbean.

1.1.4 Climate change in Vietnam

Located in South-East Asia, Vietnam is a tropical monsoon country of which agriculture is the main sector of economy. Agricultural and forest area account for 77% of country area with 50,178,378 tonnes of cereal in 2014 (FAO estimate [54]). The country is in the top 10 of the most affected by impacts of extreme weather events in terms of fatalities as well as economic losses from 1996 to 2015 (Global Climate Risk Index 2017 [99]).

⁴www.fao.org/ag/locusts

⁵http://faostat3.fao.org/browse/G1*/E

⁶<http://www.un-redd.org/>

According to Ministry of Natural Resources and Environment of Vietnam [120], one probable scenario shows mean sea level rise on coastal area (or line) of Vietnam will be 25cm in 2050 and 73 cm in 2100. Other scenarios give mean sea level rise on coastal zone of Vietnam higher than global mean sea level rise. In Vietnam, the most affected region due to climate change is Mekong Delta.

Mekong Delta, Western Region (Vietnamese: Mien Tay) or the South Western Region (Vietnamese: Tay Nam Bo), is a southern region of Vietnam, including one big city (Can Tho) and 12 provinces (Long An, Tien Giang, Ben Tre, Vinh Long, Tra Vinh, Dong Thap, An Giang, Kien Giang, Hau Giang, Soc Trang, Bac Lieu and Ca Mau). The delta, which produces nearly half of the country's rice, is experiencing impacts of climate change which is expected to be worse by the time passing. These effects include:

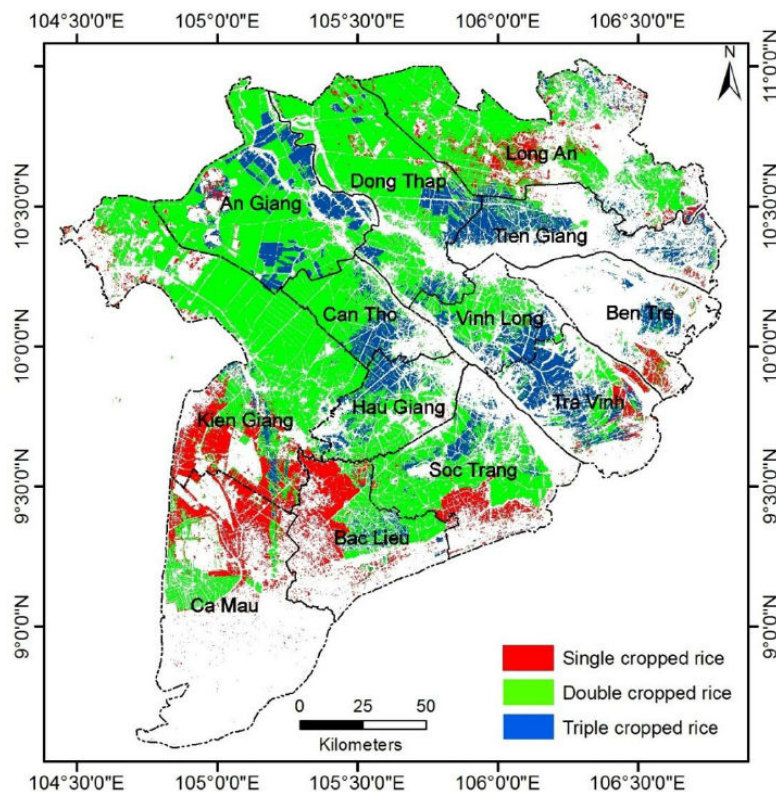


Figure 1.1: Mekong Delta - A rice region of Vietnam: its rice crop map derived from Envisat ASAR WSM data from 2007 (taken from [123]).

- Floods. They appear during the rainy season, from May to December. Indeed, living with floods to take their benefits [39][124] is an adaption strategy that

government and citizens have applied. However, after heavy raining, the low elevation land in the delta gets long inundation. For example, in Can Tho city, the largest city in this region, daily maximum inundation depth, is around 10cm on street now, it should reach 70cm in 2050 [163].

- Sea level rise. From 1993 to 2010, data measured from satellites shows that mean water level for the entire coastal zone of Vietnam increases approximately 2.9mm/year where it tends to rise more strongly in Mekong Delta [120]. It is able to impact inundation projection by increasing base level [173]. And coast line should be shift up in 2030 and 2100 of 25 km and 50 km respectively in the lands [201].
- Drought and salinity intrusion. From 2015 to 2016, the whole country and particularly the delta have suffered from a historical salinity intrusion [192] causing drought emergencies in seven on thirteen provinces of the region and salt water intrusion which has been extended 20-25km further inland than seasonal averages due to particular high sea water level.

Insect pest is another problem due to its strong growth and high dissemination which impact the environment and agricultural activities since decades. The growth of those particular insect species is facilitated by good weather condition (hot temperature) particularly favored by the climate change. Recent projection of climate change urge to tackle this issue. Combination of climate change and insect pest can be disastrous. That question is investigated to illustrate a framework supporting climate change modeling and simulation.

1.2 Rice insect threat in climate change condition

1.2.1 Study of Brown Planthopper threat

The brown plant hopper (BPH), *Nilaparvata lugens* Stål (Hemiptera: Delphacidae) is a major insect pest of rice in Mekong Delta, South Vietnam. This type of insects causes big loss of rice production by sucking rice leaves and transmitting two virus diseases: Rice Ragged Stunt Virus disease (RRSV) and Rice Grassy Stunt Virus (RGSV) disease [23], spread over the delta, resulting fall of rice production. For example, from 2006-2007, rice, particularly in the Mekong Delta, suffered a major setback due to BPH which has

caused the loss of 348,927 ha in 2006 and 572,419 ha in 2007 [25].

Characteristics of BPH as any living organism are defined by two aspects its capacity to multiply and to move what can be summarized in two models: one based on reproduction and the other on propagation [125][137][150].

Reproduction model of BPH

A reproduction model tries to define capacity of a living organism to live enough to be able to produce descendants (i.e., to multiply the species). All in all, life cycle of BPHs extends on 26-30 days [150] depending on environmental factors and it is distributed in three phases: eggs, nymphs and adults. Growth time lapse of each phase is as followed: eggs 6-8 days, nymph 12-15 days, adults 19 days (figure 1.2).

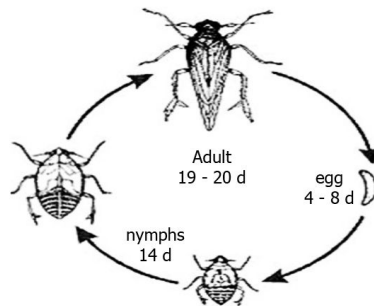


Figure 1.2: Three stages of development of BPHs. ⁷after hatching

Reproduction of BPHs is controlled by meteorological conditions such as temperature, humidity, precipitation, and wind as well as other factors such as rice maturation, insecticide concentration or its natural predator [78][125][82].

Impact of rice maturation on BPH development (Effect of rice on BPH proliferation)

Rice, is the main food of hoppers, goes through three phases in its growth [151]: vegetative, reproductive and ripening phases. It takes, normally, three to six months from seeds to be matured plants depending on its varieties and environmental factors. Usually, it is divided into two groups: long variety which matures in 150 days and short variety which matures around 120 days. The three phases of the rice growth are:

⁷(Source: <http://agropedia.labs.iitk.ac.in/agrilore/sites/default/files/BPH%20rice.pdf>)

- Vegetative phase. It starts at germination and ends just before panicle initiation about 52 days after sowing. The number of days in this phase depends on the variety of rice, normally from 55-85 days.
- Reproductive phase. It is from the developing panicle to flowering. This phase is about 30 days and is almost the same for long-duration varieties as well as short-duration varieties.
- Ripening phase starts at flowering and ends when the grain is mature and ready to be harvested. It takes place about 30 days for almost tropical varieties but may be 65 days in cold countries. This phase can be subdivided into milky, dough, yellow, ripe, and maturity stages; normally their characteristics are based on the texture and color of the growing grains.

The best conditions for the development of BPHs are young rice, developing panicle and milky stage.

BPH main predator

In the wild BPHs have predator, natural enemy, which is a factor to control them. In fact, these enemies are able to eat and transmit diseases to kill hoppers. The major predator for BPHs is *Cyrtorhinus lividipennis* Reuter (Hemiptera, Miridae) [152] which has a life cycle of less than 20 days (table 1.1):

Table 1.1: Three life cycle of *Cyrtorhinus lividipennis* Reuters.

| Stage | Name | Days |
|--------------|--------------|-------------|
| 1 | Eggs | 6 |
| 2 | Nymph age 1 | 3 |
| | Nymph age 2 | 3 |
| | Nymph age 3 | 3 |
| | Nymph age 4 | 3 |
| 3 | Male adult | 17-19 |
| | Female adult | 17-19 |

Pesticide

Usage of phytosanitary (chemical) product had been increase since the last 50 year. It is a factor which is able to control BPHs. Actually, pesticides are chemical compounds derived from nature or chemical synthesis and used for pest and disease control. Some pesticides just influence BPHs while others can kill predators as well. Impacts of those products on farmer health force them to reduce their usage. Resilience those molecules allow them to be found in rice produced and in the soil years after.

Propagation model of BPH

When conditions are unfavorable, BPHs tend to propagate to other places. Indeed, young rices are their favorite food, hoppers tend to stay on those fields to eat until food is depleted. When rices become mature or ripe, with the assistance of the strength of wind, adults equipped with long wing are able migrate to other fields to find new food resources. Wind can advect hoppers to far away places, however, without it, they can also propagate to near fields with their own velocities, approximate 0.4m/s [31].

Under some weather conditions, BPHs can propagate to very far destinations (500km up to 750km) [130]. Therefore, radar systems can also be used to observe those migrations. Observations using high frequency radar in China [153] shows that long distance return migrations, causing by northeastern wind, occur in mid and late September. It also found that hopper clouds take off in the late afternoon or at dusk, fly at heights between 400 and 1000m above ground level (AGL) under air temperature around 16°C. Other observations [145] recognize that flight durations are about 9-10h at heights below 1800m AGL in summer and below 1100m AGL in autumn. Those migrations are caused by northeastern wind as well.

1.2.2 Manual measurement system: Light trap

Light trap is a device which is able to attract insects to its light bulb due to their phototaxis. From insects collected in the device, their species and densities are calculated, therefore, trap acts as measure device, or a classifier/counter.

Light trap (figure 1.3) is considered as an useful tool in pest management. For instance, in Britain, Rothamsted light traps [36] have been established since the early 1960s in order to understand insect population change during pest and to measure and analyze their diversity [203]. In addition, to confront with Brown Planthopper (BPH)

impact, light traps have been used in Mekong Delta [25] so people can better know situations of their fields and take the best decision.



Figure 1.3: Picture of manual insect light trap taken in O Mon, Can Tho.

Table 1.2 depicts densities of BPHs collected in Dong Thap, Vietnam in 01/2011. Normally, light traps are operated around 4 hours per day and insect densities are calculated in the next morning.

Table 1.2: Densities of BPHs in light traps collected in Dong Thap, Vietnam from January, 1st to January, 19th 2011. Green cells depict the maximum of insects caught in a day of over light traps.

| | | | | | | | | | | | | | | | | | | | |
|---------------|-----|-----|-----|-------|-----|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|-------|-------|
| MAX | 432 | 320 | 430 | 1,395 | 940 | 2,470 | 1,500 | 2,650 | 1,350 | 13,780 | 22,710 | 17,340 | 28,450 | 26,800 | 72,181 | 31,300 | 18,700 | 6,170 | 1,780 |
| Sa Rai | 0 | 0 | 12 | 0 | 31 | 9 | 13 | 2 | 16 | 0 | 131 | 36 | NE | 4,237 | 3,950 | 25 | NE | 1,250 | 1,370 |
| Tan Thanh A | 106 | 73 | 85 | 103 | 128 | 97 | 85 | 78 | 93 | 325 | 11 | 1,250 | NE | 3,455 | 2,570 | 982 | 2,518 | 950 | 545 |
| Long Thuan | 185 | 93 | 210 | 198 | 95 | 215 | NE | 324 | 390 | 410 | 450 | 515 | 5,200 | 3,500 | 2,950 | 2,100 | 895 | 1,800 | 410 |
| Thuong Phuoc | 25 | 35 | 18 | 29 | 20 | 44 | 93 | 322 | 870 | 2,030 | 1,342 | 1,870 | 3,830 | 4,270 | 4,370 | 5,120 | 4,150 | 2,130 | 1,780 |
| Tan Hoi | 290 | 320 | 215 | 86 | 56 | 70 | 46 | 76 | 256 | 320 | 637 | 322 | 1,235 | 1,600 | 1,273 | 967 | 3,650 | 1,700 | 1,655 |
| Phu Tho | 30 | 16 | 26 | 16 | 0 | 48 | 390 | 340 | 460 | 658 | 560 | 740 | 7,522 | 8,928 | 4,228 | 3,480 | 2,320 | 1,146 | 540 |
| Phu Cuong | 19 | 22 | 19 | 13 | 0 | 21 | 42 | 38 | 33 | 964 | 1,250 | 1,338 | 8,640 | 7,980 | 5,790 | 2,546 | 2,034 | 187 | 58 |
| Tan Binh | 7 | 0 | 2 | 6 | 0 | 4 | 6 | 3 | 0 | 18 | 20 | 19 | 3 | 13 | 6 | 8 | 16 | 23 | 0 |
| Thanh Binh | 6 | 3 | 5 | 0 | 0 | 10 | 18 | 11 | 6 | 0 | 0 | 0 | 0 | 0 | 4 | 11 | 3 | 5 | 7 |
| My Tho | 78 | 126 | 430 | 1,395 | 940 | 2,470 | 123 | 237 | 740 | 13,780 | 22,710 | 17,340 | 28,450 | 26,800 | 17,200 | 31,300 | 18,700 | 6,170 | 232 |
| Phuong Thinh | 37 | 52 | 7 | 14 | 11 | 24 | 22 | 47 | 14 | 17 | 34 | 48 | 427 | 1,247 | 1,386 | 936 | 426 | 418 | 98 |
| My Hiep | 10 | 12 | 15 | 11 | 9 | 13 | 8 | 14 | 12 | 9 | 6 | 11 | 54 | 420 | 329 | 2,011 | 102 | 68 | 49 |
| Gao Giong | 96 | 30 | 0 | 19 | 0 | 0 | NE | 0 | 0 | 0 | 0 | 96 | NA | NA | 72,181 | 214 | 322 | 580 | 1,200 |
| My An | 0 | 0 | 0 | 0 | 0 | 2 | 325 | 1,200 | 650 | 127 | 1,150 | 500 | 1,600 | 6,300 | 2,350 | 370 | 29 | 13 | 7 |
| Doc Binh Kieu | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 18 | 25 | 57 | 32 | 95 | 230 | 800 | 1,600 | 760 | 130 | 170 | 40 |
| Truong Xuan | 0 | 9 | 0 | 36 | 0 | 1,200 | 1,500 | 2,650 | 1,350 | 3,200 | 2,400 | 12,000 | 13,200 | 15,000 | 7,500 | 500 | 3,200 | 2,200 | 670 |
| Lap Vo | 4 | 22 | 11 | 9 | 15 | NE | 11 | 64 | 36 | 19 | 12 | 3 | 2 | 4 | 5 | 26 | 8 | 4 | 11 |
| Tan My Binh | 86 | 142 | 52 | 93 | 226 | 63 | 42 | 218 | 55 | 26 | 208 | 64 | 682 | 916 | 184 | 956 | 472 | 62 | 138 |
| Thanh Trung | 432 | 235 | 181 | 269 | 106 | 97 | 118 | 86 | 69 | 18 | 74 | 102 | 328 | 935 | 1,263 | 642 | 58 | 44 | 75 |
| Dinh An | 31 | 52 | 27 | 124 | 5 | 16 | 62 | 52 | 41 | 45 | 23 | 62 | 124 | 46 | 121 | 94 | 76 | 30 | 41 |
| Long Hau | 7 | 5 | 3 | 4 | 2 | 5 | 0 | 12 | 14 | 9 | 5 | 18 | 25 | 240 | 280 | 340 | 164 | 58 | 36 |
| Phong Hoa | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 6 | 2 | 1 | 7 | 4 | 250 | 11 | 2 | 25 | 4 | 8 | 6 |
| An Phu Thuan | 8 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 2 | 36 | 22 | 83 | 862 | 1,824 | 2,086 | 4,283 | 6,224 | 1,006 | 726 |
| Tan Phu Trung | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 43 | 56 | 103 | 124 | 86 | 32 | 37 |
| Phuong An Hoa | 0 | 0 | 13 | 29 | 11 | 5 | 0 | 2 | 76 | 59 | 45 | 36 | 5 | 16 | 0 | 7 | 4 | 5 | 2 |
| My Tan | 0 | 0 | 4 | 5 | 0 | 67 | 38 | 103 | 76 | 138 | 2,400 | 840 | 10,600 | 13,800 | 6,112 | 420 | 390 | 312 | 264 |

However, these samples as table 1.2 are retrieved manually. Therefore, an automatic device to categorize and number insects would be a great help to avoid hand-operated computations.

1.2.3 How to mitigate BPHs threat : Problem statements

The case presented in sections 1.2.1 and 1.2.2 points out following problems:

- How to classify and/or calculate insect densities automatically? An automatic mechanism is able to reduce labor force in manual counting.
- Any capable solutions to interact/control insects at local sites? Interaction is defined as possibility to change insect behaviors.
- Networking all sample locations. Network can provide global information, spatially distributed, to confront with insect pest.

Actually, the goal of this thesis is to design an insect early warning system in Mekong Delta based on data collection in local stations. An early warning system is defined as *"a chain of information communication systems comprising sensor, detection, decision, and broker, in the given order, working in conjunction, forecasting and signalling disturbances adversely affecting the stability of the physical world; and giving sufficient time for the response system to prepare resources and response actions for minimizing the impact on the stability of the physical world"* (Waidyanatha's definition [198]).

eLocust3 [92] is that kind of system to survey, control locust, record data and transmit them through satellite in real time. It is composed by following functionalities:

- Regular surveillance. Handheld devices ⁸ controlled by field officers are able to manually record current situations including ecological factors, insect conditions, pesticide use. Firstly, environmental parameters such as habitat type, soil moisture, vegetation conditions and state, rainfall are mandatory storage for both operationally and for basic research purposes. If locusts are present, their statuses, comprised of type of infestations (hoppers, bands, adults, swarms) and their stage, density, size, colour, behaviour and activity, are entered by employees. Next, control operations are also considered for later evaluation of treatments.

⁸<http://www.fao.org/ag/locusts/en/activ/DLIS/eL3/index.html>

- Data transmission. Collected data is transmitted via satellites (Inmarsat ⁹) to centralized locust control centre within a matter of minutes.
- Analysis and forecasts. Models and simulations are done based on GIS (Geographic Information System) analysis to get forecast insect migrations, rainfall, locust eggs and hoppers estimations.

However, data collection in eLocust3 is done manually with the assistance of hand-held devices. The insect surveillance in the thesis, in the other hands, contains automatic sites to categorize/count insects and transmit numeric data to a centre where simulations are done to estimate the current situation and future trends.

1.3 System methods to confront BPH

1.3.1 Modeling and simulation: A key point

As shown in section 1.1.3, several initiatives were led about insect invasion modeling, classification/counting and even technical issues for cooperation between different simulations on different platforms.

Definitions of model can be given as:

- *A model is a representation of an idea, an object or even a process or a system that is used to describe and explain phenomena that cannot be experienced directly* ¹⁰.
- *Computer model is a representation of a system or process created on a computer, to assist calculations and predictions* ¹¹.

Models, which present insect invasion, are categorized in two types: continuousness and discreteness.

Continuous models

Insect invasions can be described as continuous models by using differential equations. Rolling swarms of locusts [181] are introduced to model the insect migration by using

⁹<https://www.inmarsat.com/>

¹⁰Source: <https://www.sciencelearn.org.nz/resources/575-scientific-modelling>

¹¹Oxford dictionary: https://en.oxforddictionaries.com/definition/computer_model

numerical simulations and tools from statistical mechanics, the notion of H-stability. This kinematic model incorporates social interactions, gravity, wind, and the effect of the impenetrable boundary formed by the ground. The rolling structure is similar to the ones observed by biologists, including all locust activities from takeoff, landing zone, and how they rest and feed. Mathematical models [44], which depict how swarm cohesion can be maintained by huge population of insects over long distances and long periods of time, are proposed by using travelling wave solutions. Those models focus on interactions among organisms in an insect swarm and are mostly concerned by forces which hold the swarm together once it is formed.

Discrete models

Locust outbreaks can be expressed by discrete modeling. A tool for insect dynamic simulation based on cellular automata ¹² [115] is designed to model spatially and temporally desert locust population dynamics in relation with geographic area. A physical system, place where desert locusts behave, allows performing synchronous simulation with space and time discretization. Simulations are concurrent and can be executed on multi-core processors or graphic accelerators. Locust population evolves after a certain number of synchronous turns can be seen with respect to insect life cycles. Stochastic cellular automaton model of locust population-dynamics on lattices [97] is introduced to model the gregarious phase and interactions among individuals of insects. Simulations are made by changing environmental parameters of the model to clarify the relation between the weather data and insect behaviors.

BPH behaviors under environmental factors [137][125] are modeled and simulated by using agent based approach in GAMA [172]. The first work about BPH invasion modeling is depicted in [137] where wind is the main factor for describing hopper behaviors. The wind data is a parameter simulation scenarios, it is not taken from an outside sources likes meteorological stations. In addition to wind, other factors such as temperature, humidity are also considered to model BPH reproduction [125]. Common things in those studies are the weather data which is not taken from a meteorological source and the calculation of the simulation is done based on administrative area (districts, cities).

BPH behaviors and light traps are combined into a model to optimize the light trap network for monitoring invasions of insects [184]. Light traps are also considered as

¹²to appear in 2017 Winter Simulation Conference (WSC)

part of model to produce best estimation of insect populations. Perception zone and neighbor zone of a light trap are proposed and those are similar to sensing range and communication range of a sensing device.

1.3.2 Insect classification and counting

Solutions for estimate insect densities are varying from imaging or acoustic method and are able to run on or offline. The common objective of these solutions is to free people from manual counting which is particularly time consuming and leads to fatigue and an increase of error rate.

Image processing is a common method to count insect densities [146][118][41]. The idea is to distribute a camera system in order to capture insect images, next some segmentation methods are used to extract insect images from the background. Background subtraction is the most common used, however, some other methods (e.g Seeded Region Growing) may be used to increase reliability of the segmentation phase. Some features including color, shape and texture features of each pest are extracted into a classifier for classifying and counting. Nevertheless, authors in [41] use a pipeline based on deep learning, part of a broader family of machine learning methods, for identifying and counting pests in images taken inside field traps. This solution relies more on data, not on human knowledge since it uses no pest-specific engineering which enables it to adapt to other species and environments with minimal human effort.

Photoacoustic sensor can be used to classify insects thanks to their wingbeats [12][139]. The logical design of the sensor is described in figure 1.4. It is composed of a low-powered laser source and a phototransistor. An electronic board is connected to the phototransistor. The laser is pointed at a total internal reflector which returns the slightly scattered light back to the source, with some of it hitting the phototransistor. When a flying insect crosses the laser beam, its wings partially occlude the light, causing small light fluctuations captured by the phototransistor. These signals are filtered and amplified and they are treated as audio signals.

1.3.3 Simulation and multi-simulation solution

Simulation represents behaviors of a system which is described as a model while multi-simulation relates to interplays between several ones. The High Level Architecture (HLA) [79][100][80], a standard for distributed simulations, is used for interoperability

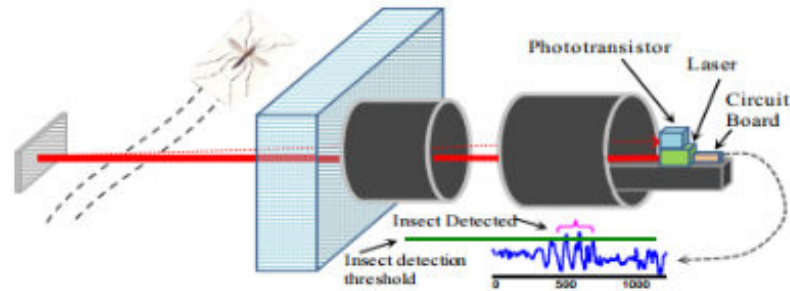


Figure 1.4: Diagram of photacoustic sensor (from [12]) used to recognize insects.

and reusability in context of latter. In HLA terminology, the whole problem which needs to be modeled and simulated is represented by a federation. Each simulator (sub models) referring to the federation is called a federate. A set of federates is connected via a Run Time Infrastructure (RTI). In this case, a RTI can be considered as distributed operating systems for cooperation of federates (figure 1.5).

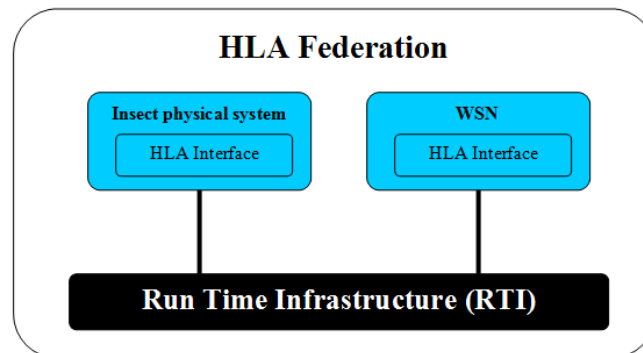


Figure 1.5: HLA federation architecture used in multi-simulation.

Federates exchange data together via RTI. For example, a federation has two federates of which the second one relies on values provided by the first one. Initially, the first federate publishes its states to RTI and the second needs to subscribe them. Whenever the second receives states from RTI, values of state variables in its simulator are updated to make suitable behaviors. However, the second federate publishes its states but the first one does not need to subscribe.

Besides, both federates need to register synchronous points (figure 1.6) to synchronize their data as well as activities. This can be done by defining a protocol in HLA Time Management Services.

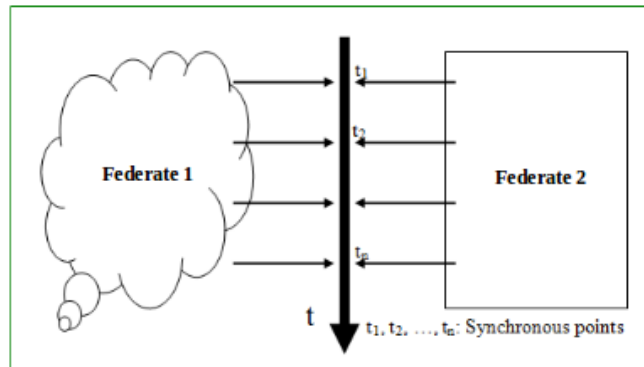


Figure 1.6: Synchronous points in a federation.

1.3.4 Methodology: Sensing, understanding, controlling

To our knowledge, it seems that there is no complete solution from local practical automatic observations at sampling locations to global system for modeling and simulating insect invasion in a small or average region (or area). Therefore, this dissertation focuses on that kind of such solutions.

Objective of this dissertation is to design of sensing machine to understand threats as well as measure and control counteract factors. It is applied on the case study in section 1.2 so:

- Old light trap is replaced by automatic sensing device which frees farmer from manual classification and numbering.
- New devices are connected to networks to feed collection of data used in insect or environment simulation.

Sensor is a device which detects or measures a physical property and records, indicates, or otherwise responds to it ¹³. Their distribution is able to measure surrounding conditions to give values of the environment at sampling locations. It also plays as actuators to activate environmental factors. The two ways relation between these equipments and the environment is one of the research topic of this thesis.

Sensing devices are deployed to constitute a whole system to monitor natural phenomena in a defined area. Components of the system are able to provide global picture of physical world to face climatic events.

¹³Oxford dictionary (<https://en.oxforddictionaries.com/definition/sensor>)

1.3.5 Thesis layout

The rest of the thesis is organized as followed:

- Chapter 2 illustrates a methodology to observe automatically the surrounding conditions by the design of vision sensor architectures and application. It is about sensing aspect, thereby, the local scale interaction is able to be found here. Using sensors local scale interaction can be defined and complete a cyber physical modeling.
- Chapter 3 deals with method to consider the physical world as a machine on which processes are executed synchronously and presents an information system solution for that machine. The proposed solution is able to integrate data of different types from different sources as well as allow data sharing via multiple simulations. Global decisions are probably made when observing historical data through years.
- Chapter 4 presents a study case merging model insect behaviors under environmental factors developed in chapter 3 and insects monitoring based on vision sensing developed in chapter 2. Variables used are wind, temperature, humidity, precipitation to assess capacity of environmental model to provide an overall view of the insect pest (or to predict behavior of insect pest).
- Chapter 5 presents conclusions of work done in the dissertation. Future plans are drawn to extend analysis of local and global sensing system.

2

A vision observatory: insect sampling

2.1 Automatic observatory - Basic principles

Observatory is best known as a room housing astronomical telescope or other scientific equipment for the study of natural phenomena ¹. In our context, it is understood as a station equipped with sensing devices to study environment. Another example of observatory is a site built to observe boat activities on a wide surface to notify drivers whether boats are in their right tracks to avoid collisions. If both monitoring and notification are done automatically, it becomes an automatic observatory.

World Meteorological Organization (WMO) provides some clarifications with automatic weather station. It is defined as meteorological station at which observations are made and transmitted automatically [209]. Therefore, automatic station (or automatic observatory) is a site equipped with sensing devices to locally diagnose phenomena

¹Oxford dictionary: <https://en.oxforddictionaries.com/definition/observatory>

happening in field of biology, chemistry, geology and physics and to send results to destinations via automated transmissions.

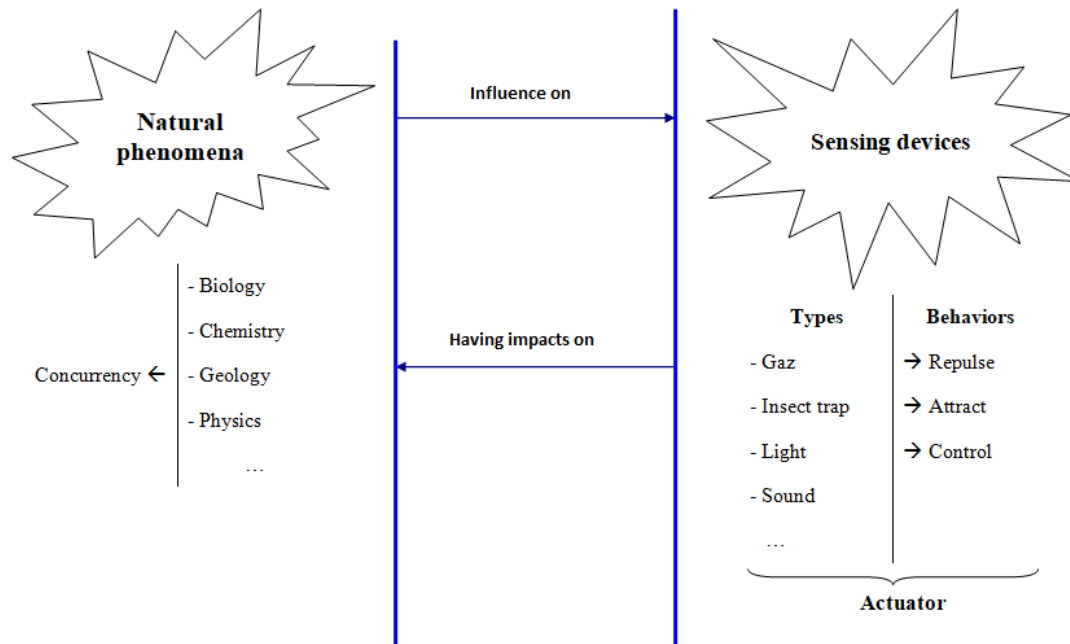


Figure 2.1: Diagram of local automatic diagnostics showing relation between natural phenomena and sensing devices.

Sensing devices are able to measure phenomena values at a defined timestamp (or time step). For instance, sensors are equipped with hydrological meteorological station to compute regular weather parameters (know as factors in computer application), water level and water flow in Vietnam [121]. The typical frequency of measurements is four times per day. Vietnamese hydrological meteorological station, using time series, the trend is sampled and plays as an early warning system of floods for habitants in a region. Another example is the insect trap (also know as bio-diversity trap) which monitors behaviors of insects and other arthropods. It is able to reduce their populations by attracting them to a light bulb and destroying. Their population numbers are processed to better understand situation of crops.

In addition, those devices are able to cause repulsing. That means they have impact on natural phenomena. Ultrasound is used as a sound trap [90][5] which is able to control bats activities.

As a result, the presence of an observatory induces interactions between natural phenomena and sensing equipments as shown in figure 2.1.

In the second example, actuator in the trap is the main factor able to change insect behaviors, thus, it is popular use in agriculture (as in [86][17]). Actuator is defined as a person or thing which can make a machine or a device operate ². Indeed, light bulb in a light trap stimulates insects due to their phototactics. In fact, the structure of the compound eyes of insects influences their spectral sensitivities. It is the case of yellowish devices with planthoppers, leafhoppers, aphids, whiteflies and leafminer flies [194][46].

Bio-diversity trap is a sensing device having its own sensing range. For example, in the case of light trap, sensing range relies on several factors, from trap itself to lunar phases [101]:

- Although luminous intensity of artificial light is theoretically constant, but change of voltage may modify some parameters such as life cycle, power input, and light efficiency.
- Natural illumination (time and span of twilights, Moon phase, light pollution) may be added to intensity of the light bulb as well. This illumination may be calculated upon geographical position, season of the year or during one night.

According to Nowinszky L. et. al [101], the sensing range of a light trap can be calculated as:

$$r = \sqrt{\frac{I}{ES + EM + EN + ELp}} \quad (2.1)$$

Where

- r: sensing radius (m).
- I: luminance of the lamp, measured in cadela (cd).
- Denominator: Natural illumination consists of the sunlight (ES), the moon (EM), the starry night sky (EN) and the light pollution (ELp). For example, the sensing radius for 125W mercury vapour lamp [18] is 35m at a Full Moon, 518m at a New Moon. Another example [43] is a 125W HPL light source which has the sensing radius of 70m at a Full Moon and 830m at a New Moon. They are calculated in lux (lx) equivalent to lumen per square meter (lm/m^2), flux of light on the surface of the trap.

²Oxford dictionary: <https://en.oxforddictionaries.com/definition/actuate>

This chapter explains natural rules and counteract of building automatic observatory, with a special focus on vision systems. The general orientation is to set small programmable autonomous communicating systems able to do *local diagnostics* and to emit synthetic information toward remote databases. In fact, this chapter describes a methodology for automatic observatories equipped with vision sensors architecture applied on bio-diversity trap for insects categorizing and numbering.

The rest of the chapter ³ is organized as follows. Section 2 depicts architecture of an automatic observation system using vision sensor equipped with fisheye lens to observe an insect trap. The next section presents camera calibration method to estimate object size in insect trap. It also provides indications to create dataset and use it to categorize and to count insects in the section 4. Experiment results and evaluation are described in section 5. Section 6 illustrates operating principals of automatic trap including some scenarios and establishments which are able to change insect behaviors. A summary of the chapter is discussed in the last section.

2.2 Automatic vision system

2.2.1 Objectives of the visual sensor

Automatic vision system (AVS) is understood as a site uses vision devices and other instruments to observe environment and collect data locally. Therefore, it is designed as an integrated concept of various measuring devices to combine *data acquisition and processing*. It is unnecessary to transmit collected images via a network as problems concerned in [62][136]. Nevertheless, in an AVS, executions are done on site and the data is transmitted to destination. No image is maintained in the station.

An AVS can be built from following components:

- Autonomous system node. An autonomous sensor [3] is a device able to perform its task without being connected to a control unit. Emergence of application fields requires increase of computational capabilities with suitable power consumption in the sensor node. Raspberry pi [148], NVIDIA Jetson Tk1 [128] are examples of such devices that are able to deal with new challenges of application fields (figure 2.2, 2.3).

³Content of the chapter is progressed during the thesis time

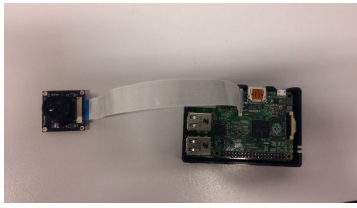


Figure 2.2: Raspberry 3.0 equipped with a camera.

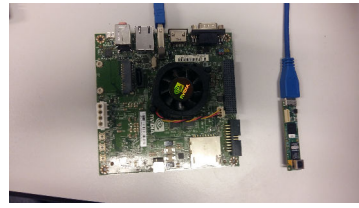


Figure 2.3: Jetson Tk1 board with an eCam camera.

- Radio link and network organization. Sensor nodes are connected each other mostly using radio connection in a network topology. By integration of radio transceiver, each sensor node is able to transmit its messages to distant destinations. For instance, Zigbee [2] can transmit a message at roughly 50m while LORA [160] is able to reach tens of kilometers at low data rate. Mobile communication (3G, 4G, 5G) is another solution to carry messages to distant destinations.
- Suitable power consumption with high performance. For example, Raspberry Pi 3.0 consumes typically 800mA, 4W in its tasks while Jetson Tk1 requires 12W in peak GFXBench 3.0 Performance Metrics.
- Sensors. Sensing station consists of sensors such as temperature, accelerator, or even more specific as acoustic and vision. Using data from sensors, sensing station is able to carry out its task locally (local processing) and transmit output to a data center via radio links.

The AVS is applied in a bio-diversity trap in Mekong Delta, Vietnam to *classify/count insect densities automatically* so that it can free farmers from *current manual tasks*. The requirement is that quantities of insects *are collected periodically* (e.g 30 minutes, 1 hour, 1 day), therefore, *realtime observation is not necessary*. The only obligation is, after a period of time, to return the values from the station. That gives the local view of insects with respect of surrounding conditions at the node.

2.2.2 Camera system and technical issues

Camera system is a kind of device which is able to be used as a sensor in vision applications. Generally, a camera system is composed by two main components: vision sensor and lens.

Vision at low level process

The key point of vision applications is the relation between the pixel matrix sensor and the processing unit. In practice, there are three types of cameras as shown in figure 2.4:

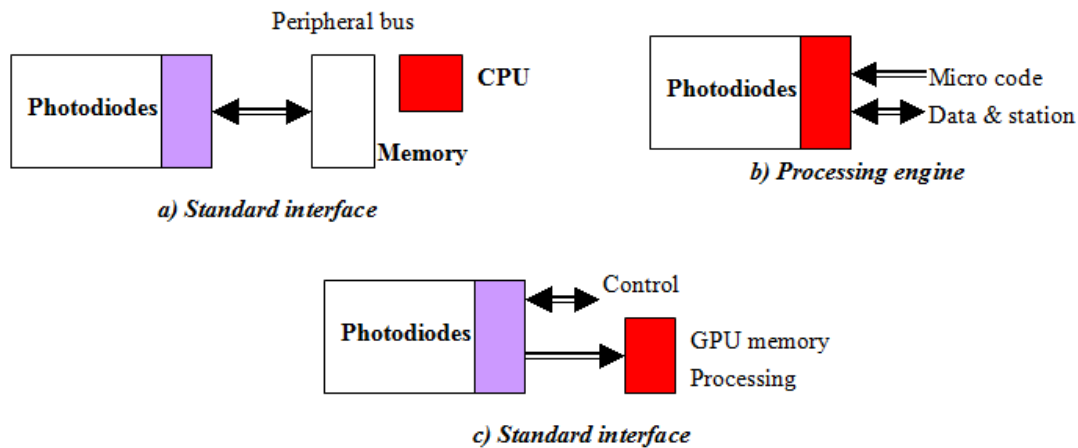


Figure 2.4: Several vision approaches: (a) standard camera separating vision matrix and processing unit, (b) smart sensor with photodiode matrix and processing on the same chip, (c) camera assembly with DMA between the vision matrix and a graphics processing unit memory (GPU).

1. Figure 2.4a illustrates the architecture of widespread cameras as used in mobile phones, for example. They are inexpensive and follow standardization of access interfaces. In principle, a camera has a sensor matrix which is isolated with a processing unit. A micro-controller can control structure of the image, resolution, acquisition speed in frames per second. This affects the quality of videos relevant to final user.
2. Intelligent cameras for industrial controls integrate the processor (a parallel processor) and the sensor array (figure 2.4b). Analysis can thus be carried out directly into the camera without rendering image format. These cameras can recognize tens of thousands of objects per second, without significant energy expenditure and they also allow development of machine learning techniques.

Figure 2.5, 2.6 depicts a prototype of that kind of camera (G. Fabregat et al. [48]). It consists of parallel processor integrating with sensor array and also registers. Via registers, commands are made to ask some lines of the matrix, then these lines are mapped to the processor for analyse. It is a kind of smart sensors where diagnostics are done locally.

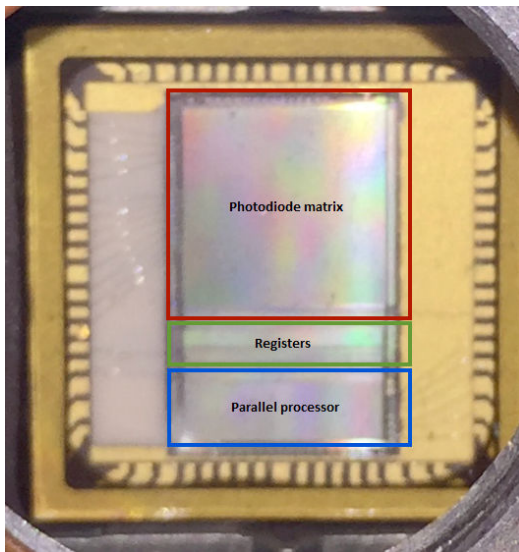


Figure 2.5: Camera integrating a parallel processor and sensor array.

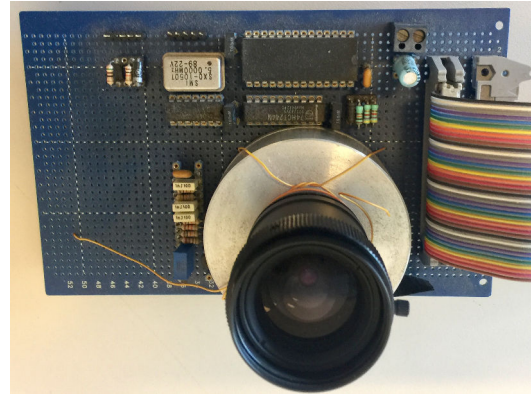


Figure 2.6: The smart camera equipped with a lens.

Camera in 2.5 and 2.6 is a 'smart optical sensor' since it has a combination of sensor and processor in the same chip. It uses MAPP2200 (Matrix Array Picture Processor) vision chip [154][6] which has 256*256 array of photodiodes to capture an image. It is able to handle a row of pixels in parallel thanks to its SIMD (Single Instruction, Multiple Data) processor. Figure 2.7 illustrates the block diagram of its integrated circuit.

Object size calculation is able to be done inside the camera using its parallel processor. Image data is read out line by line to a parallel array of amplifier. Once a line of data is mapped to the PD register, a threshold is applied to the row to emerge object pixels without using A/D conversion (Analog to Digital). Size calculation algorithm starts from first line. Operation MARK provided from the instruction set of the sensor is able to combine objects in the new line (the register) which is vertically connected to objects in the previous rows which are maintained as a row in the accumulator. Its result is stored in the accumulator (figure 2.8). Sizes of objects are kept in another register. When processing a new line, if objects in accumulator disappear (e.g third object as in figure 2.8), their sizes are returned. If they are still exists, their sizes are accumulated with sizes in the new line. The process continues to the last line, then sizes of objects are calculated. In addition, no image information are maintained.

3. Halfway between these two techniques, devices in figure 2.4c) are able to control

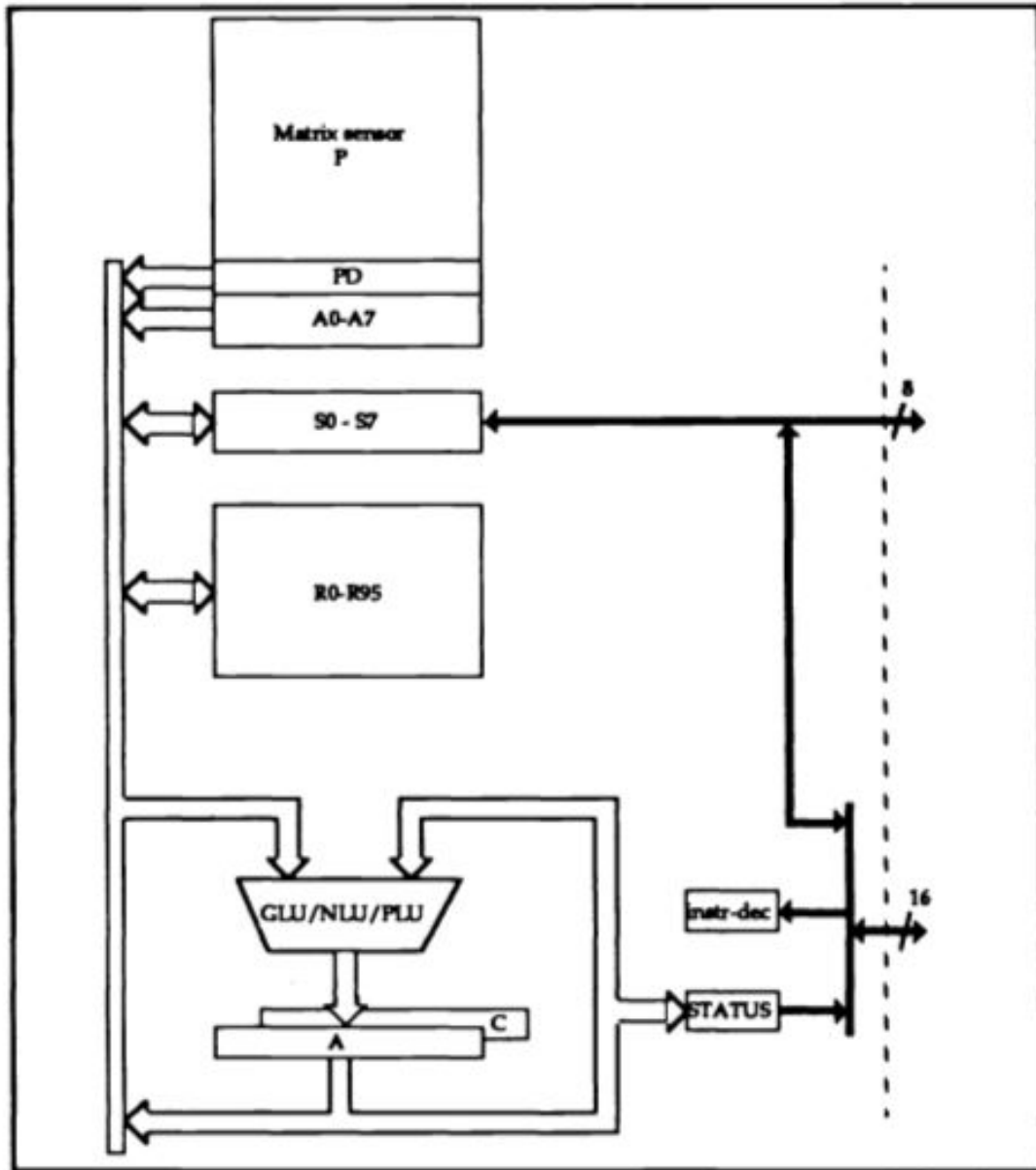


Figure 2.7: MAPP220 Block diagram (source from [154]), chip equipped with a photo matrix, 114 digital registers, an analog register and three types of pixel processing units: Point Logical Unit (PLU), Neighborhood Logical Unit (NLU), Global Logical Unit (GLU).

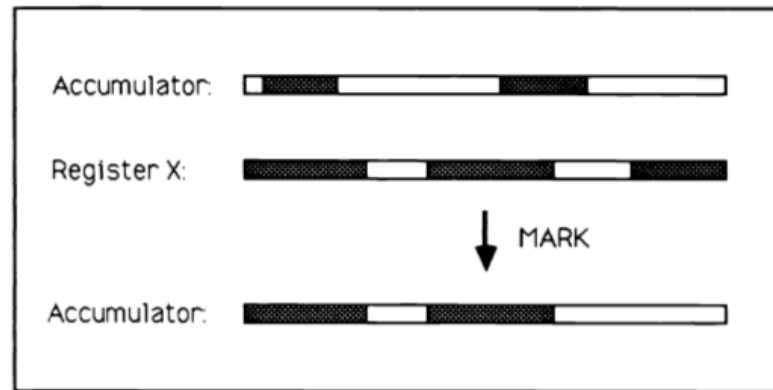


Figure 2.8: Description of object extraction using MARK operation. In the register, objects vertically connected to objects in the accumulator are preserved (source from [154]).

acquisition of image segments by sending them directly to a GPU.

What is important in concept of parallel vision (case b and c) is that pixels appear in lines, or group of lines, they are treated as a whole: there are no sequential loop over a line, but concurrent processing of several pixels. That is the result of properties of the architecture and warranty the virtualization of the array processing.

Some pros and cons of three types of camera are illustrated in table 2.1. The table uses Raspberry Pi with camera as type 1 (2.2), IVP MAPP 2200 as type 2, Jetson board with camera as type 3 (figure 2.3).

Table 2.1: Comparisons between three types of cameras.

| | Camera (a) | Camera (b) | Camera (c) |
|---|-----------------|--|--|
| Processor | Embedded CPU | Parallel processor | Embedded CPU and GPU |
| Data acquisition & processing unit | Seperated | Integrated | Seperated |
| Power consumption | 4W | Low. Because there is no energy spent for transferring data from data acquisition and processing | 12W in peak GFXBench 3.0 Performance Metrics |
| Programming | General purpose | A specific task | General purpose |
| Design | Pre-designed | Full custom designed | Pre-designed |

Depending on each monitoring application, a suitable camera type is used. For example, to observe wildlife or birds, type (b) or (c) are necessary; however, *in the insect case, camera (a) is chosen. Technically, an image is captured by the camera and micro-controllers process this image to give density values.*

Basics in optics

Because light rays are projected on sensor surface of a camera to form an image, they must pass through a lens. Amount and quality of light passing through the lens determine quality of the projected image.

In a homogeneous medium, light travels in straight lines. One example of this characteristic is pinhole camera [29]. A pinhole camera is an optical imaging device formed by closed box which has a very small hole (O) in the front face and a screen at the back (figure 2.9). If an object AB is placed in front of the hole, an inverted image will A'B' will appear on the screen.

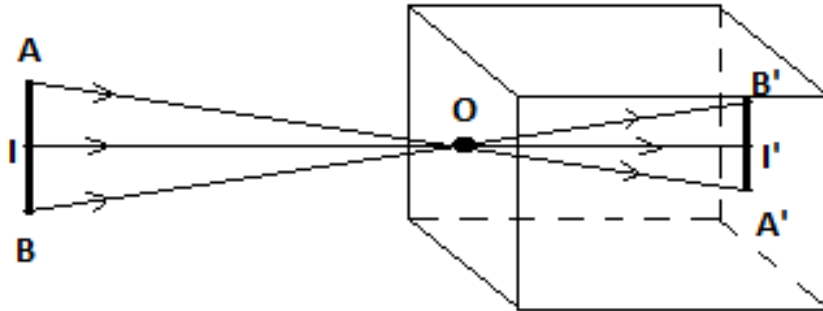


Figure 2.9: Pinhole camera with line II' is the optical axis and I' is the principal point or image center.

The size of object in real world and in image plane are related to the pinhole position by the equation:

$$\frac{\text{Image size}}{\text{Object size}} = \frac{\text{Distance of screen from hole}}{\text{Distance of object from hole}} \quad (2.2)$$

Terms relating to lens (figure 2.10) are:

- Focal length. It is the distance from the optical center of a lens to the vision sensor when the lens is focused on an object at infinity.
- Field of View. It is the angle over which the sensor can *see* through the lens.

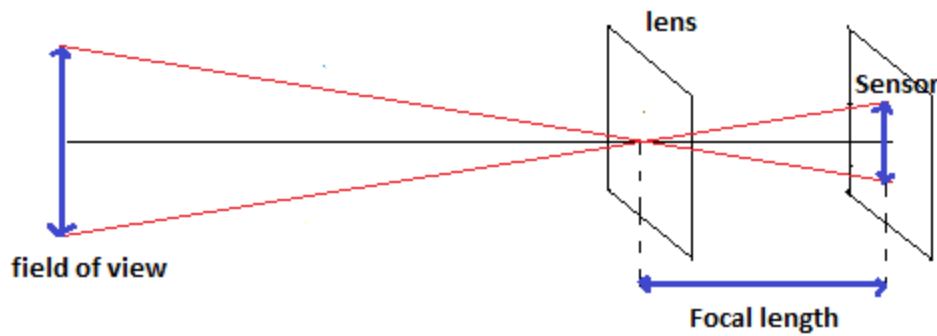


Figure 2.10: Focal length and field of view of a camera lens.

For a given sensor size, specified by the diagonal length of a single frame of sensor, a lens may be classified [200] as a:

- Normal lens. It has a focal length approximately equal in length to the diagonal.
- Wide-angle lens. It has a focal length shorter than normal. Fisheye lens belongs to this type and is able to produce a circular image on the sensor.
- Long-focus lens. It has the focal length longer than normal.

2.2.3 Observation by using fisheye lens camera

Figure 1.3 depicts a manual light trap where counting is done by hand in the next morning. During the design of new automatic device, one requirement was that structure of trap stays unchanged. Therefore, our new device is similar to the old one, only electronic equipments are added to allow local processing.

Figure 2.11 depicts a typical insect trap equipped with light bulb to attract insects. The figure shows the cone used to pick up insects, sometimes a bag is placed at the bottom of the cone to contain their bodies. The top is a solar panel to generate electricity for operating the trap. Sensors are installed under the panel while there is a box which contains electronic devices such as Raspberry Pi, Arduino, battery to control the trap.

A camera to observe the cone can be considered as a solution for insect counting. It is almost impossible to observe the whole cone surface using a regular lens camera (figure 2.12), instead, at least two rectilinear lens cameras are needed to handle that. The reason is objects (if applicable) in the cone are not on the same plane. It induces



Figure 2.11: Pictures (view) of an installed automatic bio-diversity trap.

difference between objects and camera distance. If it is assumed that they are located in the cone surface, the situation still exists since the surface is not flat as well. Depth and scale information is lost when 3D points are projected onto a 2D image with a normal camera. Thus, another camera is used to compose a stereo one so that both can compute objects in the world coordinate.

3D camera is able to keep all information in the 3D world. An alternative is to use a fisheye lens camera to take wide angles of view of image planes. From this, calibration method is executed to adapt cone surface on projected image. The latter is chosen in this thesis and becomes a practical solution in the case study.

Fisheye lens [74] is a lens to provide wide-angle view, generally from 100° to 180° . Normally, it produces a circular image which appears convex, bulging out at the center, toward the edge of the image (figure 2.12). In situations such as shooting in a confined space or requirement of wide field of view, the fisheye lens is useful. Indeed, interiors of objects such as car or plane are shot by fisheye lenses and sometimes stitched together in 360° panoramas.

To observe the whole cone surface, the camera is attached on the top of the cone to compose a structure as in figure 2.13. It is an inverse cone which has 50cm diameter large circle, 10cm diameter small circle and 35.8cm distance between the two circles. The fisheye lens camera is fixed at the center of the large circle to observe objects located

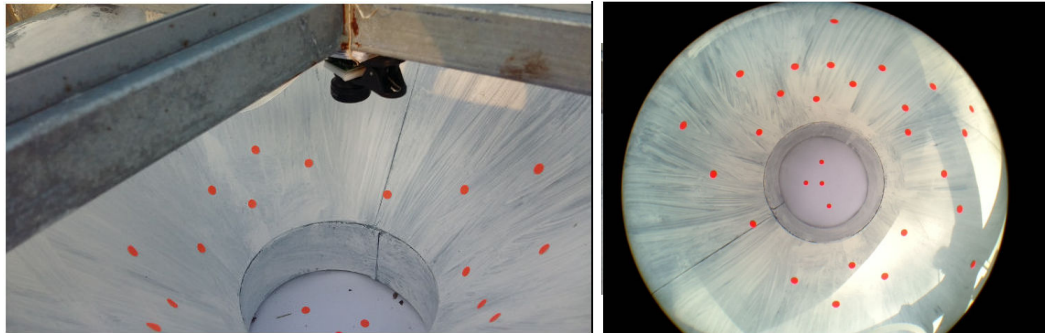


Figure 2.12: View of cone of bio-diversity trap, on the left an installed device (smartphone camera) on the right view from the fisheye lens camera

in the small circle and on cone's sides. Distance between the camera and the small circle is 34cm. Normally, insects, which are attracted by the light bulb, are falling down in the cone and sometimes slide to the small inner circle. In other cases, they are still localized on cone's sides.

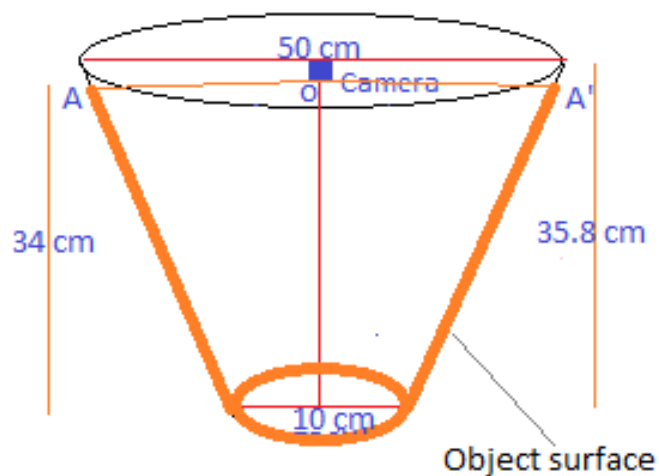


Figure 2.13: Diagram of a cone surface used in the automatic bio-diversity trap.

One problem appearing by exploitation of bio-diversity trap is how to classify and estimate insect densities. For example, from an image taken from the trap in figure 2.14, people need to know how many insects (and their species) are in the trap? Insect size becomes *a main feature to categorize species* in this dissertation because each insect

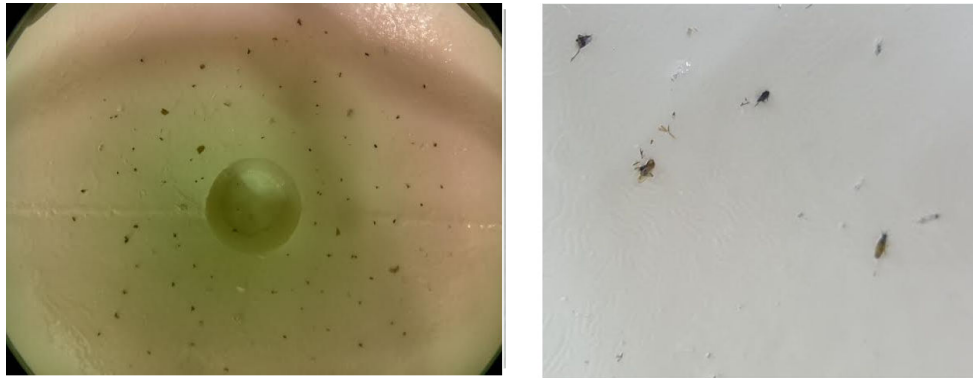


Figure 2.14: Insects caught by the trap, on the left distorted image taken by a fisheye lens camera on the right rectilinear image taken by rectilinear lens camera.

type has its own size, such as 20-30mm length in Brown Planthopper [84], 38-50mm length in Grasshopper [119]. To solve the problem, from an image captured on the cone surface of the trap by a fisheye lens camera, object size estimation need to be done.

The question is *How to evaluate the size?* It is a process done on the cone surface of the trap as in figure 2.13. It means the camera is put such as its images will have concentric center with the small inner circle. It requires engineering aspect to justify that camera and trap reach this constraint. Computer science approach is the implementation of a workflow for insect classifying/counting in a trap observed by a fisheye lens camera (figure 2.15). The workflow includes a camera calibration method to estimate object size in distorted images and application of the method to categorize and calculate amounts (figure 2.15).

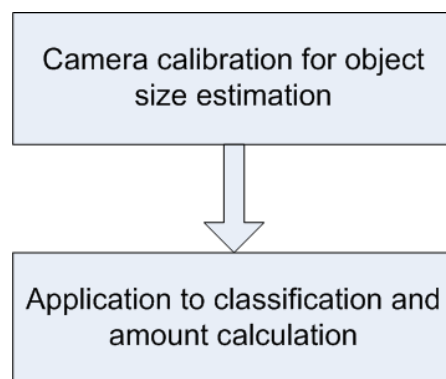


Figure 2.15: Workflow of insect classification/counting in a bio-diversity trap using fisheye lens camera.

2.3 Camera calibration in bio-diversity traps

2.3.1 Fisheye projection functions

Pinhole camera model depicts relationship between coordinates of a world point and its projection to image plane in ideal pinhole camera, where camera aperture is described as a point and no lenses is used to focus light. The pinhole projection, or rectilinear projection, preserves the recti-linearity of projected scene. It respects the function given as [122]:

$$r_u = f \tan(\theta) \quad (2.3)$$

where f is the distance between the principal point and the image plane, θ is the incident angle of the projected ray of the optical axis of the camera and r_u is the projected radial distance from the principal point on the image plane (figure 2.16).

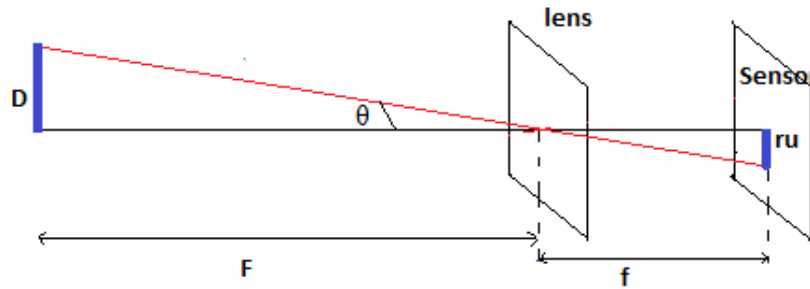


Figure 2.16: Object size projection in pinhole camera.

Formula 2.2 gives:

$$\frac{r_u}{D} = \frac{f}{F} \quad (2.4)$$

Where r_u becomes the object size in image plane, D is the object size in world coordinates, F is the distance between the object to the camera.

The fisheye projection is quite different from the pinhole projection since projected radial distance from the principal point on image plane is close proportional to the true angle of separation.

There are several fisheye projection [122] which listed as follow:

- Equidistant projection $\Rightarrow r_d = f\theta$
- Equisolid projection $\Rightarrow r_d = 2f \sin(\frac{\theta}{2})$
- Orthographic projection $\Rightarrow r_d = f \sin(\theta)$
- Stereographic projection $\Rightarrow r_d = 2f \tan(\frac{\theta}{2})$

These fisheye projections can be summarized as $\Rightarrow r_d = fg(\theta)$, where $g(\theta)$ depicts proportion of distance from the principal point on image plane and incident angle of projected ray passing through the optical center O of the camera (figure 2.17).

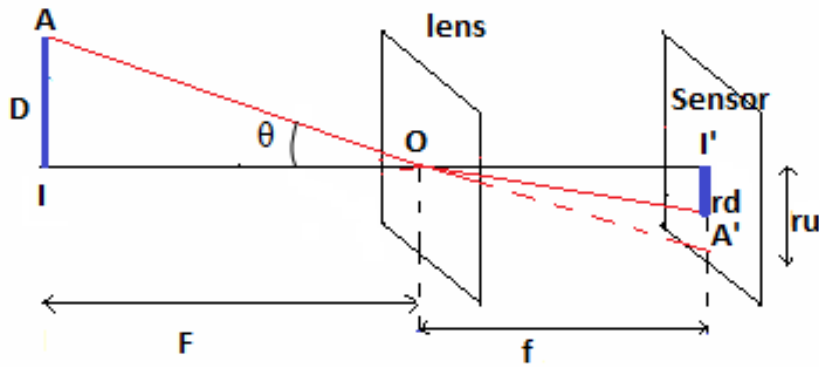


Figure 2.17: Object size projection in fisheye lens camera.

$$\Rightarrow \frac{r_u}{r_d} = \frac{f \tan(\theta)}{fg(\theta)}$$

$$\text{Thus, } \Rightarrow r_u = \frac{f \tan(\theta)}{fg(\theta)} r_d$$

(2.4) gives

$$\frac{r_d}{D} = \frac{f}{F} \frac{g(\theta)}{\tan(\theta)} = \frac{f}{F} h(\theta) \quad (2.5)$$

Similar to the rectilinear projection, r_d becomes the object size in image plane. Therefore, $\frac{r_d}{D}$ becomes the scale between the object pixels and the real object area.

Figure 2.17 shows that when the distance F between the object and the optical center is unchanged, the nearer/farther A' comes to I' (the principal point), the nearer/farther A comes to I . Therefore, an object's size is proportional to the distance between its location on image plane to image center.

2.3.2 Image center calculation

Formula 2.5 confirms that an object's size depends on its distance to the principal point. This value can be determined by using one of methods in [213][66][77].

These methods detect features from several images to estimate camera parameters. First, a pattern with some special features (corners, circles as figure 2.18) is observed by a camera with different orientations by moving either the pattern or the camera. Next the radial lens distortion is modelled and camera parameters are calculated by a nonlinear refinement based on the maximum likelihood criterion. The solution of Zhang 2000 [213] is used in this thesis to detect image center:

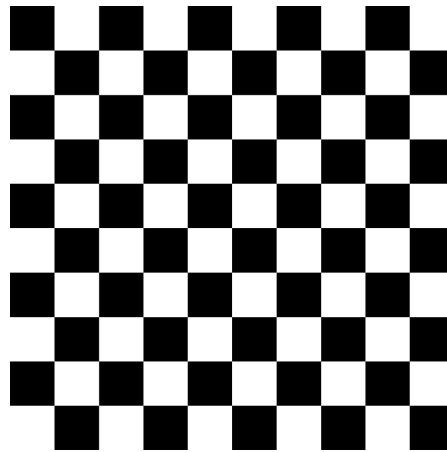


Figure 2.18: A checkboard pattern which corner points are used to estimate camera parameters.

- Print out a pattern and attach it on to a planar surface.
- Take some images of the pattern with different orientations by moving the pattern or the camera.
- Detect feature points in the pattern (corner, circle, ...).
- Estimate internal and external parameters of the camera.
- Estimate coefficients of the radial distortion.
- Refine solution using least square method.

2.3.3 Size object estimation

Description of vision in fisheye

A fisheye lens camera is attached to a light trap as in figure 2.12, it can take image of the whole cone surface. The camera is put such as the principal point of the camera coincides with the center of the bottom circle and OO' becomes axis of symmetry of the cone surface. Insects captured by trap are either located on bottom or on sides of the cone. As a result, camera calibration, which allows object size estimation, is divided into two sub cases: bottom and side.

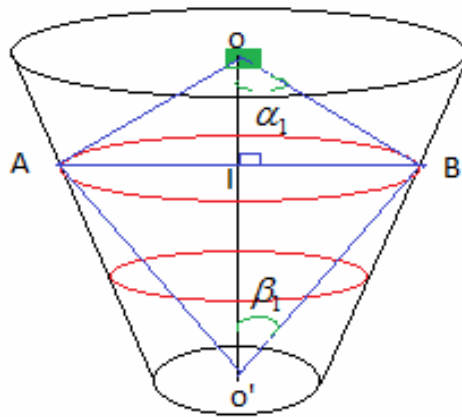


Figure 2.19: Diagram of objects located on cone sides of bio-diversity light trap with camera places at O and principal point coincides with the bottom center.

Objects in cone bottom

When objects are localized at the cone bottom, it means objects are on a flat surface which is parallel with the camera lens. Formula 2.5 confirms that object's image size relies on its distance to image center. Technically, it is necessary to identify bottom surface from cone image as in figure 2.12. From that, it is easy to check whether an object image belongs to bottom surface or not. Actually, bottom of the cone has circle shape, therefore, it becomes an ellipse in the image taken from fisheye lens camera. An ellipse detector is used to detect this condition.

Objects in cone sides

Figure 2.19 depicts the case where these objects A, B are located in edges of the cone. Suppose that A, B in the same flat surface which parallels with the camera. Image sizes of A and B have linear relations with their distances to the center I of the flat surface. However, position I can move from O to O'. It means that image sizes of objects A or B rely on $|\vec{IO}|$ or $|\vec{IO}'|$.

In implementation, \vec{IO}' is easy to be identified because O' is just the center of the ellipse imaged of the cone bottom.

Moreover, $IO' = O'A \cos \beta_1$ (figure 2.19)

Therefore, image sizes of any objects A (B) depend on O'A (O'B) or on the distance between each object image location and the center of the cone bottom (*the principal point*).

Workflow of camera calibration

Workflow for camera calibration in a light trap to estimate object size is illustrated in figure 2.20. In this workflow, the first step is to estimate camera parameters, especially the image center (as in section 2.3.2)[213]. Next, contours of known objects in the image are identified to calculate their pixel sizes. From that, scales between those sizes and real sizes can be computed for later usages.

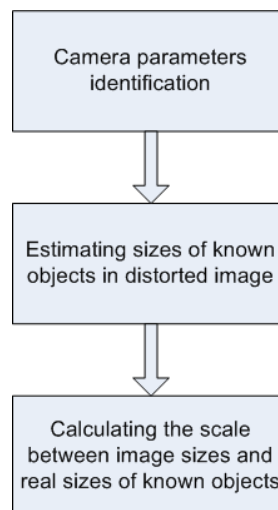


Figure 2.20: Workflow of camera calibration process used to estimate sample object sizes in a light trap.

2.3.4 Validation of calibration method

The workflow in figure 2.20 assumes that object's image size relies on its distance to image center. In other words, if objects have the same distance to the image center, then their image sizes are almost the same. In practice, it is necessary to validate this assumption, therefore, an experiment, which is done independently to the workflow in figure 2.20, is set up and some statistical methods are used for testing it.

This experiment allows the measure of image sizes of known objects in a distorted image. In this case, known objects are distributed on a plane surface and they almost have the same distance to a center (as in figure 2.21). A fisheye lens camera is set up such as distorted image center coincides with the center of the plane surface. Next some detector methods are applied to estimate object sizes in image (pixels) and the scale between these sizes and real sizes.

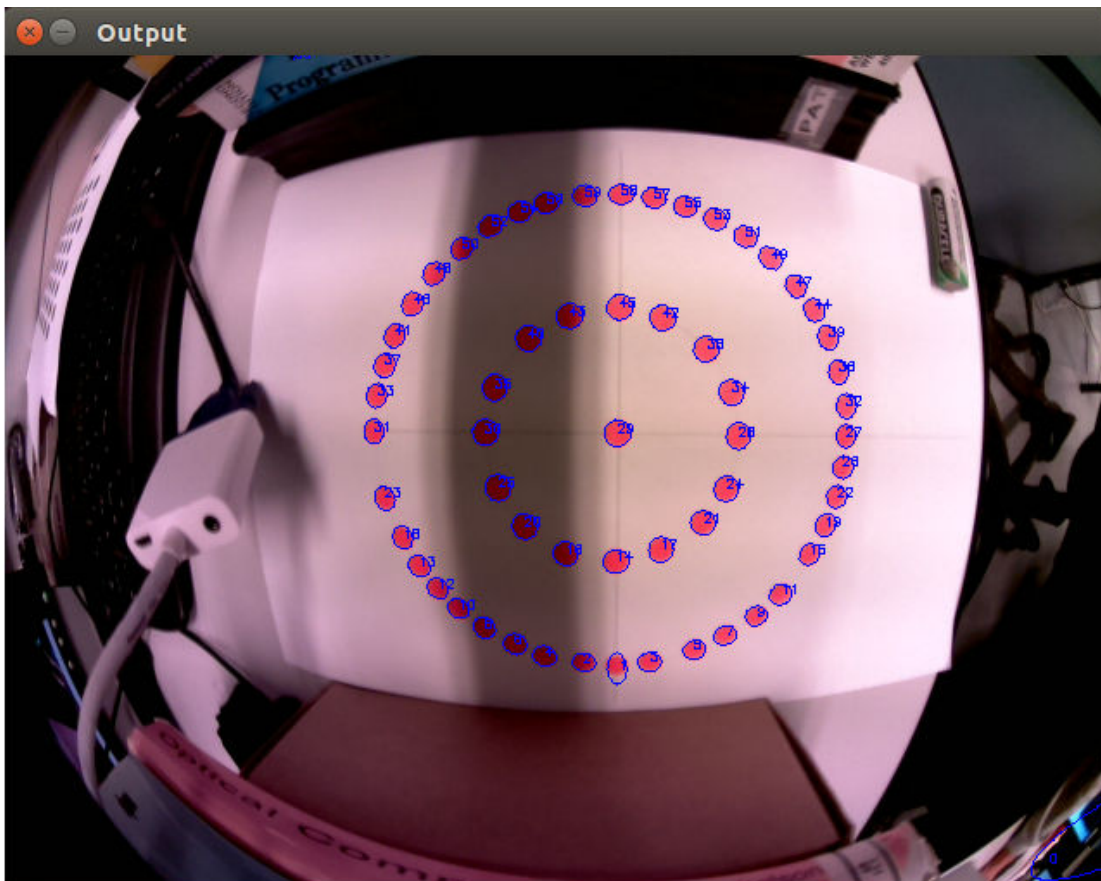


Figure 2.21: Picture of pattern of concentric circle composed by 4mm radius objects used to test distant assumption.

In order to estimate image object sizes, as example in figure 2.21, the method in [213] is used to identify camera parameters. After adjusting camera position, images of the plane surface, which consists of objects, are taken. Ellipse detectors are applied in these distorted images to extract objects' contours and compute their scales.

Raspberry Pi Camera Module with Fisheye Lens is also used in this experiment and some results are obtained as follow:

- Images are rendered as 640x480 size and the image center is estimated at (317.868, 261.45).
- Tables 2.2 and A.1 (in appendix A.2) depict scales and distances to image center of objects detected as in figure 2.21.

Scales between object sizes in the distorted image and real sizes follow normal distribution. Indeed, figure 2.22 and A.2 (appendix A.2) depict some statistical results of objects in group 1 and group 2, respectively. These results obtained from R-cran software show probability density function, quantile-quantile (q-q) plot and some tests (Shapiro-Wilk and Kolmogorov-Smirnov - appendix A.1) of object sizes in the image. For example, probability density functions represent almost bell curves while points of scales fall along straight lines of quantile-quantile plots. Moreover, Shapiro-Wilk and Kolmogorov-Smirnov tests do not reject the hypothesis of normality. To sum up, in distorted images, scale between object's pixel size and real size is the ratio with the distance between its image and the image center.

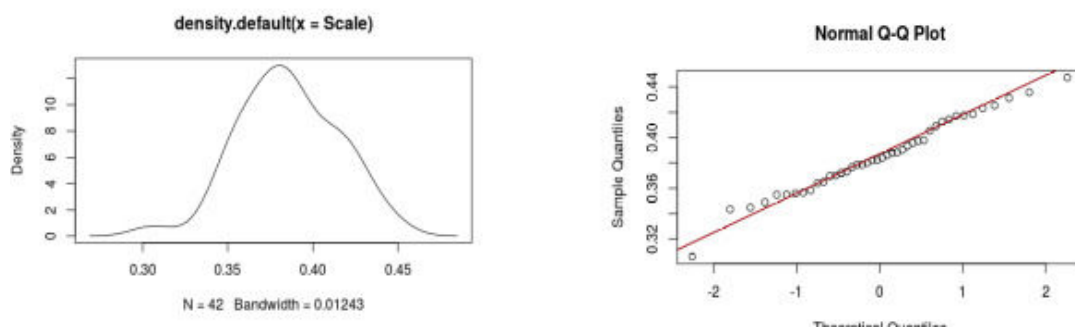


Figure 2.22: Result of statistical analysis of objects size of group 2 : on left probability density function on right Normalized quantile-quantile plot.

The Kolmogorov-Smirnov normality test which is applied on these measurements gives the test statistic $D = 0.061399$ and the significance level $p_value = 0.9564$.

Table 2.2: Expectation and deviation of measured sizes (pixels and scale) of objects in group 2 (distance around 136.32 pixels from the image center).

| Object | Distance from center | Pixels | Scale (pixels/mm²) | Object | Distance from center | Pixels | Scale (pixels/mm²) |
|---------------|---------------------------------|---------------|--|---------------|---------------------------------|---------------|--|
| 15 | 131.419 | 121.297 | 0.414399 | 1 | 136.393 | 164.163 | 0.306192 |
| 19 | 131.754 | 127.757 | 0.393446 | 8 | 136.470 | 126.495 | 0.397373 |
| 22 | 132.003 | 121.802 | 0.412682 | 55 | 136.944 | 133.423 | 0.376737 |
| 26 | 132.051 | 126.956 | 0.395928 | 10 | 137.146 | 112.363 | 0.447348 |
| 27 | 132.253 | 120.514 | 0.417092 | 12 | 137.659 | 115.364 | 0.435712 |
| 36 | 132.577 | 131.468 | 0.382339 | 57 | 137.817 | 134.716 | 0.373121 |
| 7 | 132.659 | 118.189 | 0.425297 | 58 | 138.240 | 140.941 | 0.356643 |
| 11 | 132.813 | 126.323 | 0.397911 | 13 | 138.294 | 122.908 | 0.408968 |
| 9 | 132.847 | 120.096 | 0.418543 | 18 | 138.382 | 124.015 | 0.405318 |
| 5 | 133.206 | 118.777 | 0.423192 | 59 | 138.681 | 145.799 | 0.344759 |
| 32 | 133.370 | 135.041 | 0.372224 | 56 | 139.456 | 144.054 | 0.348936 |
| 47 | 133.719 | 128.748 | 0.390418 | 23 | 139.862 | 131.524 | 0.382176 |
| 39 | 133.831 | 132.739 | 0.378678 | 50 | 139.922 | 135.816 | 0.370101 |
| 3 | 133.928 | 120.388 | 0.417530 | 54 | 140.093 | 141.599 | 0.354985 |
| 2 | 134.142 | 116.573 | 0.431194 | 46 | 140.392 | 138.081 | 0.364030 |
| 44 | 134.205 | 129.595 | 0.387865 | 37 | 140.459 | 140.261 | 0.358370 |
| 49 | 134.542 | 141.582 | 0.355028 | 52 | 140.731 | 146.393 | 0.343359 |
| 51 | 135.710 | 130.880 | 0.384057 | 41 | 140.791 | 135.979 | 0.369656 |
| 6 | 135.740 | 132.201 | 0.380220 | 48 | 140.797 | 141.234 | 0.355901 |
| 4 | 135.788 | 130.133 | 0.386262 | 31 | 141.080 | 129.413 | 0.388412 |
| 53 | 136.096 | 137.696 | 0.365047 | 33 | 141.131 | 132.763 | 0.378612 |

Experiment results confirm one more time that an *object's pixel size depends on its distance to the image center*.

2.4 Application of the camera calibration method

2.4.1 Linear regression

In statistics, linear regression [59][210] is an approach for modeling the relationship between a scalar variable y and independent variables denoted x . For example, it can be related distance between an insect and the cone center to its pixel size in light trap using a linear regression model.

A simple linear regression has the equation $y = a + bx + e$ where x is the independent (explanatory) variable and y is the scalar (dependent) variable. The slope of the line is b and a is an intercept while e is the error.

A common method to fit the regression line is the least squared method. It calculates the best fitting line from observed data by minimizing sum of squares of the vertical deviations from each data point to the line.

For example, table 2.2 shows an association between the object scale and the distance between the object to the image center. The regression equation $Scale = -0.004452 * Distance + 0.992450$ shows this relation and it is graphically represented in figure 2.23.

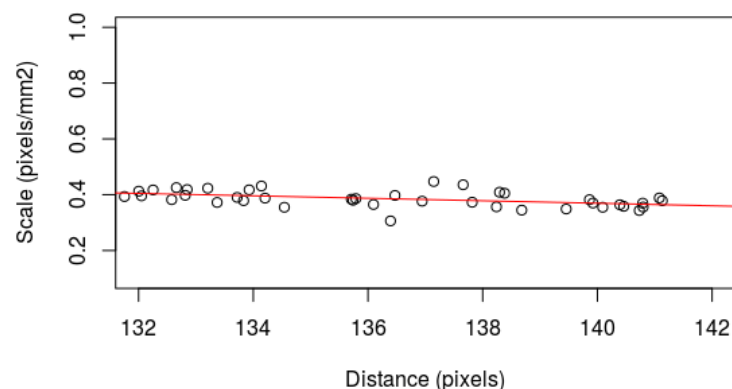


Figure 2.23: Object pixel scales relate to distance to (the cone) center, the line represents the linear regression.

Least squares method [144] is a method to approach solution of overdetermined systems by minimizing the sum of squared residual (a residual: the difference between an observed value and the fitted value provided by a model).

In simple linear regression, this method finds a, b such that $\sum_{i=1}^n (y_i - (a + bx_i))^2$ reaches the minimum value. After transformation, coefficients a and b satisfy the condition and are given as:

$$\hat{b} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.6)$$

$$\hat{a} = \bar{y} - \hat{b}\bar{x} \quad (2.7)$$

Where \bar{x}, \bar{y} are mean values of X, Y. \hat{a}, \hat{b} are estimation of a, b (not a, b) since a, b can not be calculated exactly. After that, one can estimate y_i for each x_i as below:

$$\hat{y}_i = \hat{a} + \hat{b}x_i \quad (2.8)$$

where \hat{y}_i depicts the mean value for each x_i and the rest $(y_i - \hat{y}_i)$ is the residual. The variance of residual can be estimated: $s^2 = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n-2}$

2.4.2 Insect size dataset

An insect size dataset is created by collecting insect species, distribute them to the trap surface, taken images and process these images. This phase uses camera parameters and referenced object scales from previous phase (*Sample object scale estimation* in workflow 2.15). Contour of each object is drawn and extracted to get number of pixels that the object occupies in the image. Linear regression is done to identify scale of the object based on its distance to the image center and the usage of the referenced object scales table. Object scale and number of pixels give an estimated size of the object.

For example, figure 2.24 presents a distorted image obtained from the insect trap in the creation phase of dataset. Each insect contour is extracted and calculated its distance to the image center (e.g: d1, d2). From the distance, formulas 2.6 and 2.8 are used to estimate the object's scale. Next, the object size can be computed with the assistance of the contour size and the scale ($Object_size = Contour_size * Scale$).



Figure 2.24: Distorted view of insect bodies at the surface of bio-diversity light trap cone.

2.4.3 Insect classification/counting

This phase is done in bio-diversity traps at rice fields. Everyday, insect densities in a trap are collected as presented in section 2.6 and these values are transmitted from the trap to a data center. Periodically, following steps are processed to count insects and thus compute densities:

- Take an image of the cone surface of the insect trap.
- Execute object segmentation methods. First, background subtraction is applied, and connected component labeling is used to extract object contours.
- Calculate area (in pixels) of each contour.
- Use k-nearest neighbors algorithm (kNN) and the insect size dataset to estimate quantity numbers.

2.5 Experimentation of camera calibration

2.5.1 Implementation

Implementation of this chapter is done in C using OpenCV. There are several tools involving in it:

- Camera calibration. Calibrate fisheye lens camera to retrieve the distorted image center.
- Scale estimation. Calculate scale of known objects to create a referenced size table.
- Dataset creation. Take offline images from the trap, calculate sample insect sizes and store the dataset.
- Insect counting. Count densities periodically.

These tools run on Raspberry Pi 3.0 and camera and Raspberry Pi Camera Module with Fisheye Lens.

2.5.2 Results

Camera calibration and scale estimation

The camera calibration is done in the cone of the light trap in figure 2.11. Some known sample objects (circles with 9mm radius) are put either in the bottom or on cone sides (figure 2.12). The cone image taken by the Raspberry pi camera of which fisheye lens provides image center (317.868, 261.45) at 640x480 mode.

Results of the camera calibration are depicted in figure 2.25 and table 2.2.

Counting at insect trap

Based on the referenced table 2.3, a size dataset which consists of 200 BPHs and 50 other individuals of other species are created. The dataset has some following columns: distance to image center, scale between insect image and its real size, estimated real size, and species name. Only two types of insects are considered: BPHs and others (not BPHs) since normally people pay more attention to BPHs and their impacts to rice fields.

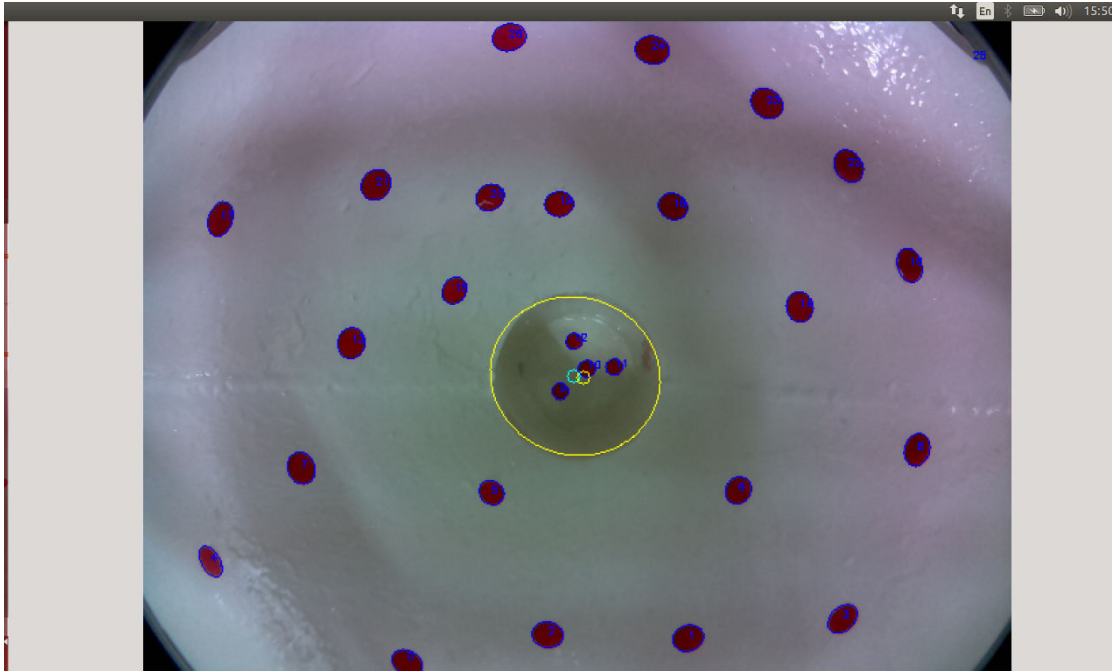


Figure 2.25: Sample objects detected in the camera calibration. The yellow ellipse depicts the cone bottom.

Table 2.3: Reference table for object sizes in insect trap.

| No | Distance to center | Pixels | Scale (pixels/mm ²) | Position | No | Distance to center | Pixels | Scale (pixels/mm ²) | Position |
|----|--------------------|---------|---------------------------------|----------|----|--------------------|---------|---------------------------------|----------|
| 0 | 261.45 | 330.891 | 0.769042 | Outside | 13 | 166.717 | 370.441 | 0.686935 | Outside |
| 1 | 209.875 | 351.979 | 0.722967 | Outside | 14 | 173.409 | 354.752 | 0.717315 | Outside |
| 2 | 191.087 | 352.8 | 0.721283 | Outside | 15 | 109.256 | 287.497 | 0.88512 | Outside |
| 3 | 265.857 | 340.521 | 0.747294 | Outside | 16 | 259.999 | 364.577 | 0.697984 | Outside |
| 4 | 301.15 | 267.75 | 0.950397 | Outside | 17 | 285.945 | 373.075 | 0.682086 | Outside |
| 5 | 105.278 | 259.14 | 0.981975 | Outside | 18 | 144.736 | 335.367 | 0.758778 | Outside |
| 6 | 146.619 | 308.268 | 0.825479 | Outside | 19 | 127.699 | 317.272 | 0.802052 | Outside |
| 7 | 212.646 | 383.929 | 0.662803 | Outside | 20 | 146.17 | 330.773 | 0.769315 | Outside |
| 8 | 257.813 | 368.956 | 0.689701 | Outside | 21 | 203.719 | 400.488 | 0.635397 | Outside |
| 9 | 15.2669 | 125.091 | 2.03426 | Inside | 22 | 254.573 | 405.898 | 0.626929 | Outside |
| 10 | 10.4587 | 130.297 | 1.953 | Inside | 23 | 245.993 | 418.818 | 0.607588 | Outside |
| 11 | 29.8484 | 125.15 | 2.03331 | Inside | 24 | 247.342 | 420.186 | 0.60561 | Outside |
| 12 | 26.2375 | 120.768 | 2.10709 | Inside | 26 | 254.688 | 404.305 | 0.629399 | Outside |

Figure 2.26 illustrates two results of counting insects in a trap in the rice field. In practice, there is no window view as in the figure, instead, after each counting, three values are returned to a Uno Arduino so the Arduino can transmit them to a data center via means of communication. The three values are the number of BPHs, the number of Not BPHs and the time that executes the counting.

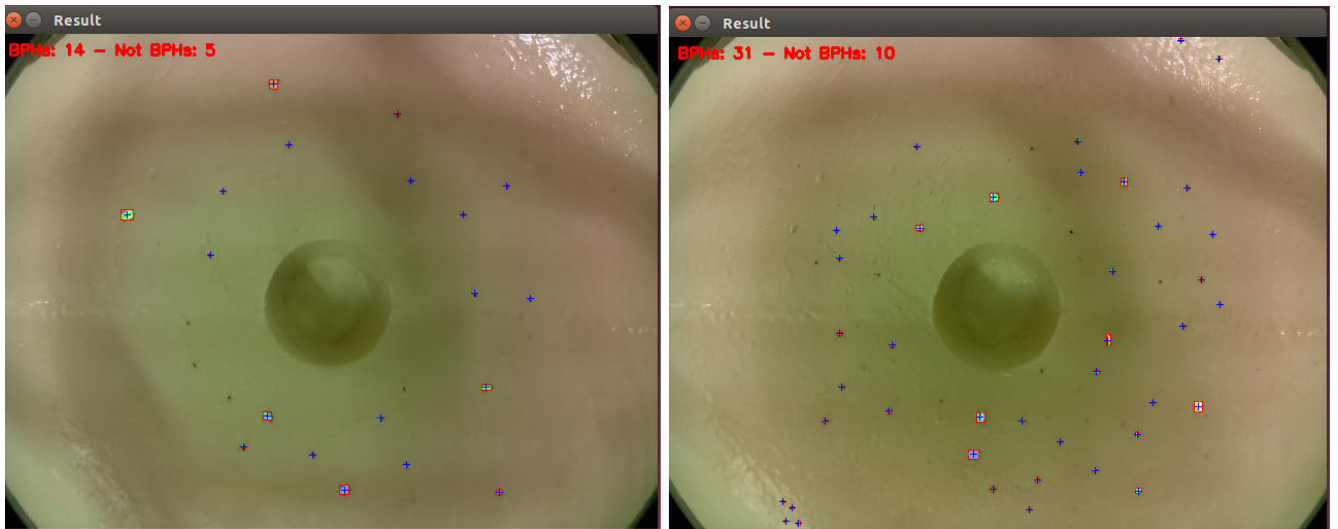


Figure 2.26: 2 results of insect counting in a bio-diversity trap.

Evaluation

The proposed method is evaluated using accuracy. It is calculated by the division of the number of true counting (TC) and the number of insects (NI) in the test data. The test data here consists of 5 images which contain 178 insect specimen.

$$Accuracy = \frac{TC}{NI} * 100\% = \frac{143}{178} * 100\% = 80.33\%$$

Some issues are addressed with the 80% accuracy. It is not a very high accuracy that is mainly because BPHs are small and produce very small image areas which make hard to identify them well. Additionally, under surrounding conditions such as: darkness, light illumination change; the quality of images may be impacted. Nevertheless, the result is generally satisfactory to reduce labor intensity of farms who should do manually previously.

2.6 Operating principals of automatic bio-diversity trap

2.6.1 Light traps network

Light traps in the same region are connected to a center to compose a network (figure 2.27). Communications between center and children (traps) include transmitting insect densities as well as meteorological parameters values from a child to the center and sending commands to change child behaviors from the center.

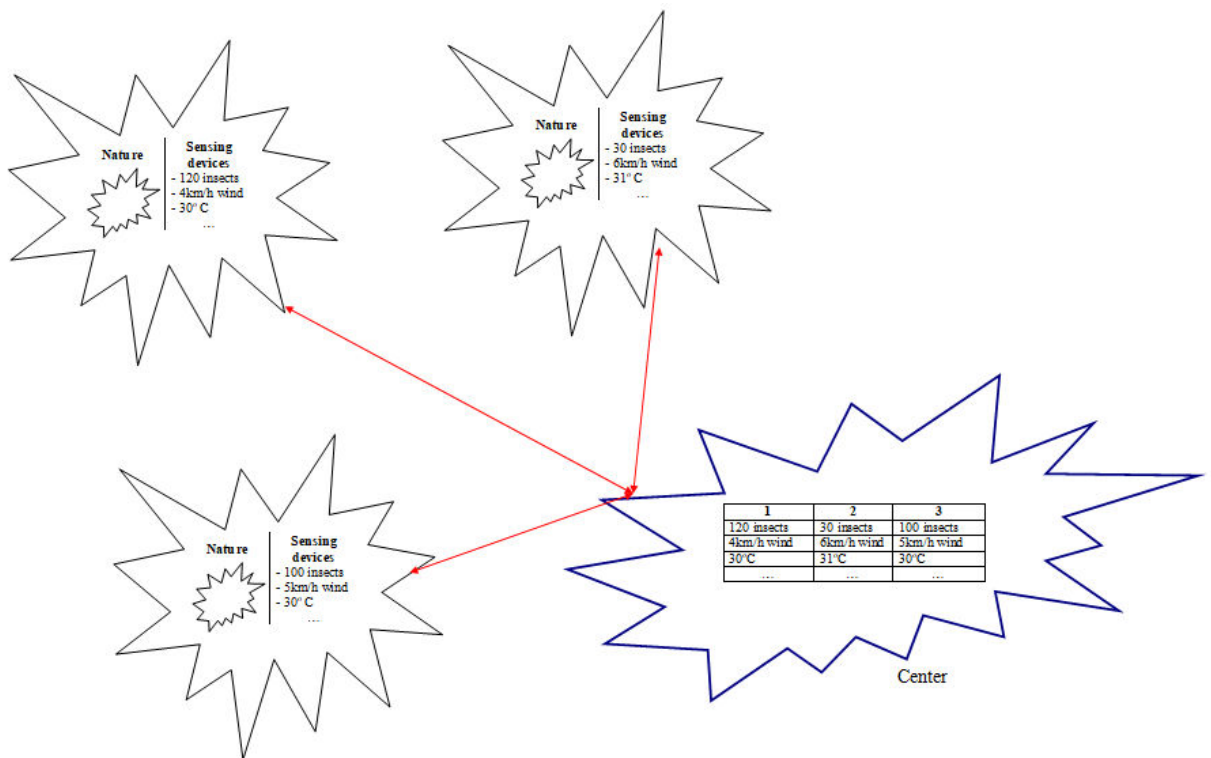


Figure 2.27: Regional network composed by three light traps connecting to a center.

That network is in test operation, it is composed of ten light traps distributed in Can Tho, An Giang, Dong Thap, three typical rice provinces in Mekong Delta. After two months operated in lab conditions, at that day (November 2017), three stations have been commissioning in Can Tho city since September, 2017. Others will be installed soon.

2.6.2 Trapping insects

Light bulbs are turned on from 19:00 to 23:00, however, this parameter can be customized depending on battery capacity and hopper level (table 2.4). The whole trap uses DC power for both fluorescent and light-emitting diode (LED) solutions. In a traditional light-trap, fluorescent lamps are commonly used due to its long time used story [11][81].

BPH behaviors may also be affected by several colors, not a lot researches have been conducted on it. However, in general, insects are able to have color receptors. A representatives study on different insects [20] found that most species possess blue receptors ($\lambda \sim 440nm$). The study also shows that most insects have green receptors maximally sensitive at $\lambda \sim 530nm$. Besides, ultraviolet (UV) receptors ($\lambda_{max} \sim 350nm$) were found and there seems no species were confirmed the lacking of UV receptors. Moreover, color sensitivity in the UV plays an important role in foraging, navigation, and mate selection in almost invertebrate animals [157]. Many insects such as BPHs operate at dusk in which blue black and ultraviolet rays overwhelm other colors.

We use the LED-based trap [14] since it allows studying the extent of BPHs attraction for different light colors: UVA, blue, green. The size and capacity of LED are compatible with those in fluorescent lamps so that they are interchangeable if necessary. Other highlights of the LED are high brightness, energy saving and long life expectancy.

2.6.3 Scenarios of trapping

Automatic sensing devices are operated in two main situations: normal night and peak night. In normal night scenario which has few hoppers, light traps gather, analyze and transmit data twice a night (at 20:00 and 23:00). In peak night scenario, they work at higher frequencies as well as exhibit more complex handling and communication than the normal one.

The difference between normal and peak night scenerios depends on the BPH level in a trap. This level is calculated based on hopper density [186] (table 2.4). Actually, values of the table were used in traditional traps to depict the damage level caused by caught insects. Nevertheless, when changing to new devices, they are reused for compatibility.

When the level changes, the rate of picture took and analysed is automatically adjusted, indicating increase or decrease of monitoring frequency. Scenario describes as follow, for example, at 20:00, the trap 1 gives 5500 hopper individuals, reaching level 4. It sends that value to the center and the center replies a command to ask the rise of the collection time interval at client sides to every 30 minutes and remaining the light on 30

Table 2.4: Hopper level related to number of insects caught by the light trap.

| Number of hoppers/trap | Level |
|-----------------------------------|--------------|
| <1000 | 1 |
| 1000 -<2500 | 2 |
| 2500 -<5000 | 3 |
| 5000 -<10000 | 4 |
| ≥ 10000 | 5 |

minutes more. At 23:30, the quantity does not exceed level 4, before switching off the light, it sends data and returns to the normal monitoring frequency.

2.7 Summary

We present a design method based on smart sensor nodes acting as cyber physical devices, *devices which are able to manage interactions between the information system and the physical aspects*, including estimation and simulation of the physical situation.

A local analysis is supported in each node with example of insect classification and counting using fisheye lens camera. It is also shown that local computation can be extended to support an architecture for high performance vision suitable to handle object recognition using parallel processor cameras.

Many applications can benefit from those integrated vision approaches, as example sound analysis.

Cyber physical characteristics can also be assessed by the possible control from sensors. It is known that physical measurements often depend on several basic parameters, and sensors may also affect those measures. For instance, it is possible to emit light signals of different colors and different intensities to probe environment. It is also possible to rotate light signal, camera, microphone or speaker. One can see that measurements, control devices, recognition form an indivisible whole which can be classified in Cyber Physic field.

3

Cellular simulation and information system structure

3.1 Introduction

Chapter 2 depicts relation between natural phenomena and sensing devices through the example of a local automatic diagnostic station (figure 2.1). This type of device can produce interactions between computing mechanisms and physical aspects. The device not only measures surrounding conditions but also emits information that can lead to modify the environment. A question is *what behaviorrs will come if these stations are combined as a whole system for monitoring environment in a broad aea?*

A global view of this combination is illustrated in figure 3.1. Local stations, distributed in a monitored area, can be connected together and compose a network that measures and controls of the environment. A database maintains all data relating with

cyber and physical interactions.

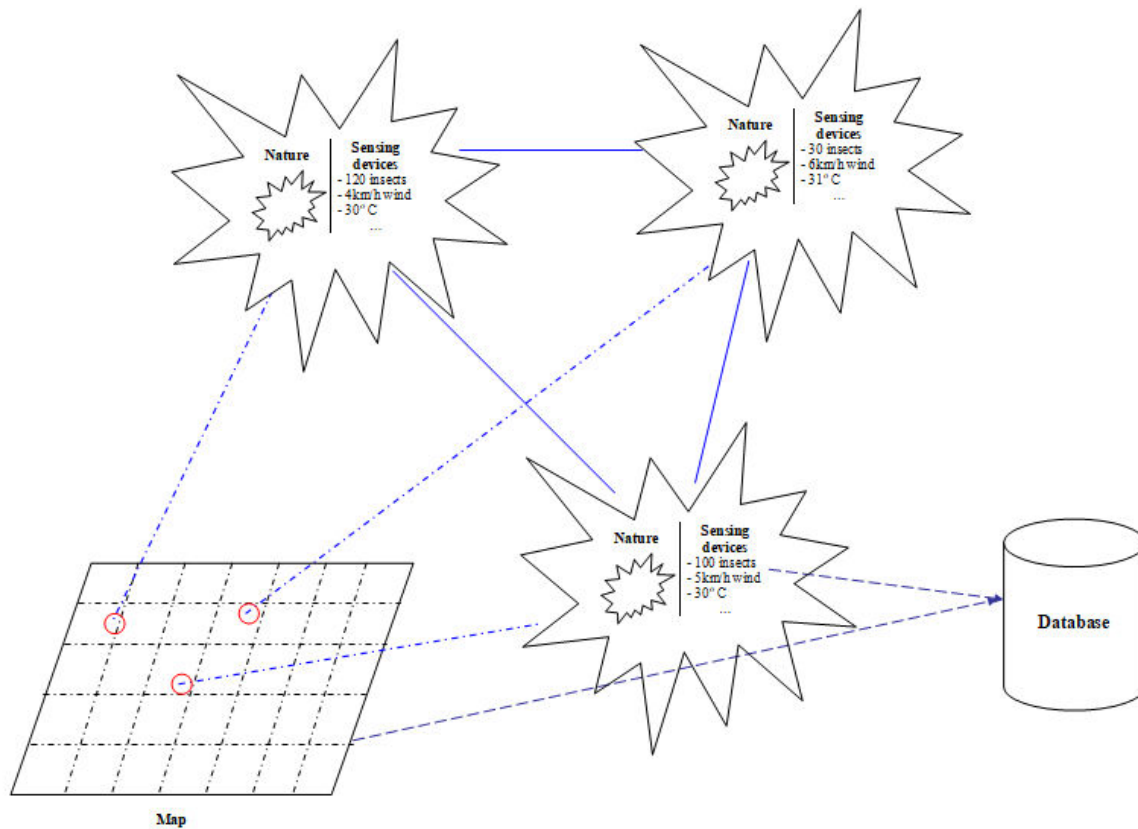


Figure 3.1: Global view of an environmental monitoring application composed by local observatory elements (or nodes) presented in figure 2.1.

Several aspects are concerned in this situation:

- **Geographic objects.** They can be called spatial objects of Geographic Information Systems. Example of geographical objects are varying from rivers, roads, tramlines to agricultural land, forest land. Most of them have their defined boundaries and other properties such as name, type, etc. For example, Mekong River has its own boundary and has Mekong as its name (property). Normally, these objects are stored in geographic database management systems such as PostGIS, MapInfo.
- **Sensor capabilities and locations.** Sensor capabilities depict attributes of the sensor such as maximum range, manufacturer, power requirements, and resolution while locations describe localizations of the sensor (latitude, longitude) (figure 3.2).

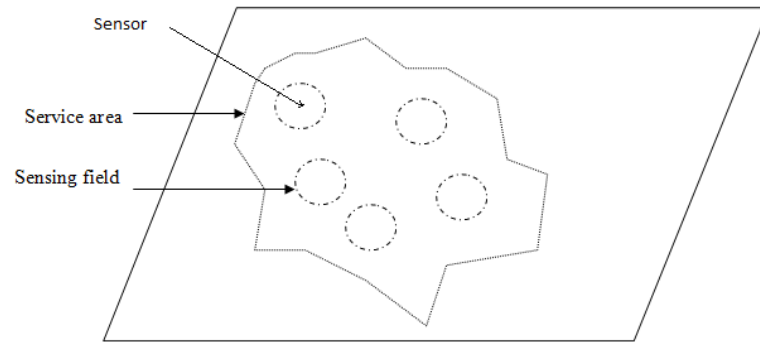


Figure 3.2: Sensors deployment to monitor a service area. Sensing field is a kind of capability to sense environment around the sensor location.

- **Timed measures.** Values of sensors at different moments. For example, table 3.1 illustrates sensing values over time in a station which is composed by 3 sensors: camera for insect counting, temperature and humidity. Localizations of sensors are changed after each timestamp as well. Figure 3.3 shows another example where weather data is obtained for the location Brest of France on September 12th 2017.

Table 3.1: Sensing values in a station over the time.

| Time stamp | Camera | | Temperature | Humidity |
|------------------|--------|----------|-------------|----------|
| | BPHs | Not BPHs | | |
| 19:00 06/09/2017 | 42 | 5 | 30.0 | 70% |
| 19:30 06/09/2017 | 50 | 6 | 30.0 | 70% |
| 20:00 06/09/2017 | 55 | 8 | 29.5 | 71% |

- Estimation of environment. From the discrete sampling on sensors, it is possible to extrapolate values for the whole service area (as figure 3.2). In other words, some estimations are done to infer the current status of the whole service area. For example, with the current data collection in light traps, what is the rice field expected density? What is the current pest situation? How to estimate other weather parameters as in figure 3.4?
- Simulation. Current data from above aspects and rules are used to predict the physical system at a timestamp in the future (as in figure 3.5). In that figure, data

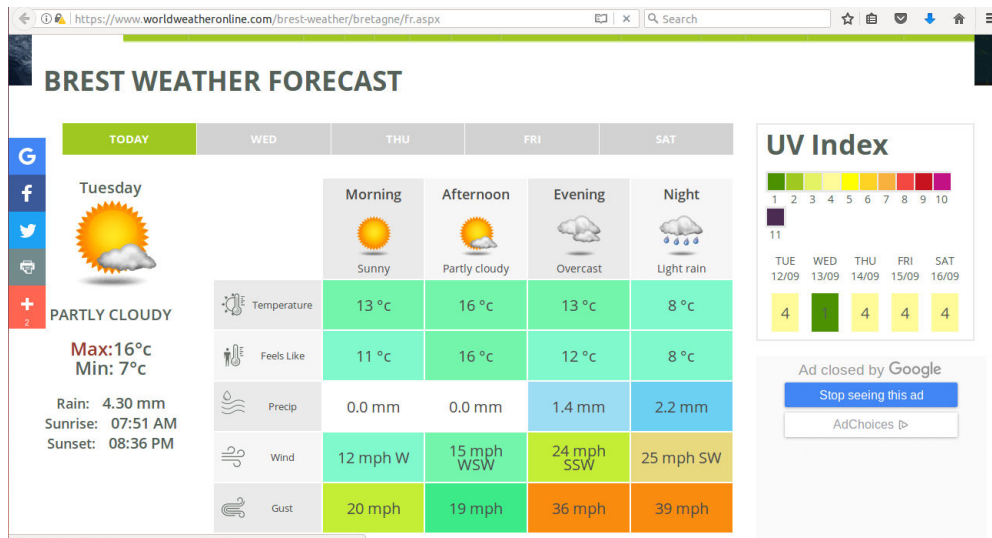


Figure 3.3: Snapshot of Brest weather forecast data of September 12th 2017 (from <https://www.worldweatheronline.com/brest-weather/bretagne/fr.aspx>).

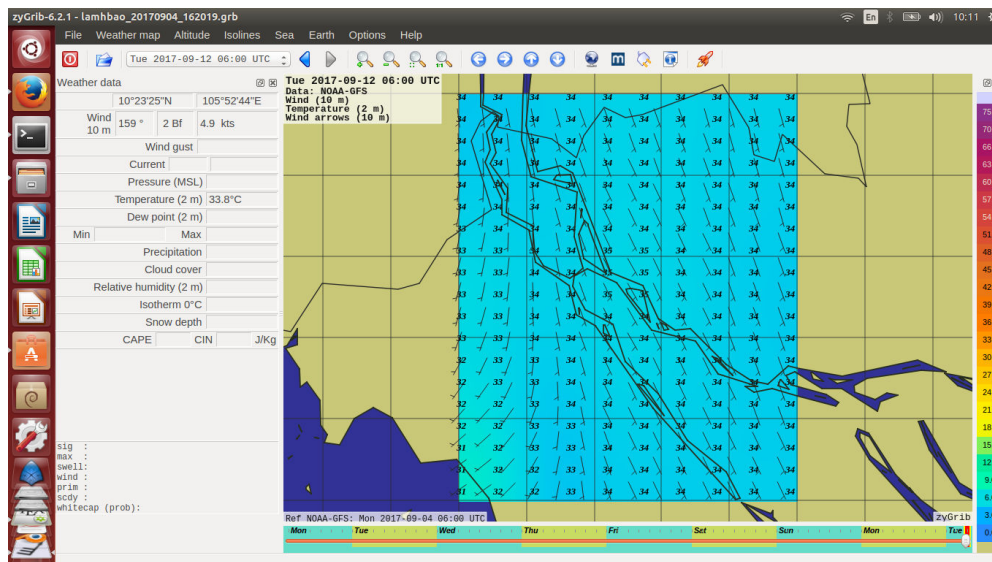


Figure 3.4: Meteorological parameters including wind, temperature at Mekong Delta in 04/09/2017 using ZyGrib [<http://www.zygrib.org/index.php?page=home>] and GFS/NOAA data [<https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forecast-system-gfs>].

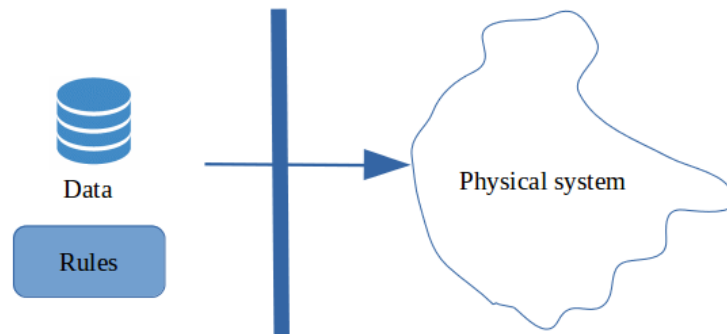


Figure 3.5: Simulation for predicting the future of the physical system.

is an element which not only participates in the current dataset but also contributes on an historical window allowing to combine with rules element to forecast.

- Simulation results. Results of simulation can be populated back to the data element in figure 3.5 in order to be used later.

Two issues appear in this context. First, what is a suitable way to model the physical system from above data aspects? For example, from data collected in light traps, what are situations of rice fields and insect pest? One solution is to divide the physical world as cells (figure 3.1) which above data aspects are mapped into. From that, rules are applied to guess next situations.

Another issue is that environmental monitoring (EM), in a general sense, requires different data descriptions and data types. For example, an insect monitoring requires weather data, light traps, insect simulation results as inputs (more on chapter 4) where a water flows monitoring may use data collection from stations, geomorphology of rivers. Obviously, each above data element is different from its descriptions and types. Therefore, a suitable database architecture needs to be introduced to adapt with these dynamic features.

This chapter presents how to model environments from local observatory elements. It also introduces an information system to tackle above questions in environment machines as presented in figure 3.1.

The rest of the chapter is as follows. Section 2 summarizes some previous work including meteorological modeling, cellular automata and sensor networks. Section 3 depicts how to model the physical system using cellular automata. Next section introduces an architecture of data access in CA sensing machine to deal with different

kinds of data. The core of this approach is a data description model that integrates data from different sources and types. Section 5 describes the implementation of the architecture where some common algorithms are introduced. Results of the proposed architecture is depicted in the section 6 while the last section presents some summary of this chapter.

3.2 Related tools and techniques

Environment modeling and the information system in this chapter inherit and develop features of following theoretical and practical aspects:

3.2.1 Modeling and simulation

Synchronous system principle

Synchronous message model [114] describes system as a synchronized series of message exchange and computation. It consists of fragment of processes which may send and receive messages simultaneously.

A synchronous distributed system can be considered as a graph G where processes are located at its nodes and these processes communicate together via their edges using message sending.

Each node in a synchronous system is termed as a process which consists of the following components:

- $states_i$: a collection of states at process i .
- $msgs_i$: a message-generation (MG) function specifies that process i sends to an indicated neighbor, starting from the given state.
- $trans_i$: a state-transition (ST) function specifies a new state to which process i moves from the current state and messages from incoming neighbors.

Cellular automata

Cellular automata (CA) are mathematical models built from adjacent elementary components. Each of these components are simple but together they are capable of complex behaviors.

A cellular automaton consists of a regular lattice with a discrete variable at each cell [204] [133]. The variable receives a value to define a state at each time step. At next time step, state is updated by executing a transition rule ϕ which is a cell neighborhoods depending function.

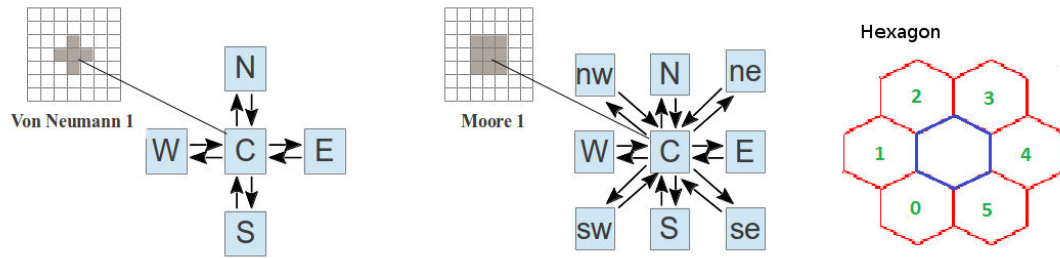


Figure 3.6: Neighborhood structure for 2 dimensional CA, from left to right: 5 neighbors square (Van Neumann scheme), 9 neighbors square (Moore scheme) and 7 neighbors hexagonal.

Complex network

Complex network derived from cellular automaton [91][180] is a technique which is commonly used to model climate data [155][49]. It is considered as a graph where cells are defined as vertices and relations between cells depict time dependent weights.

One of its application is to represent global climate system [169][170] in which raw data is distributed in grid cells representing a $5^\circ \times 5^\circ$ latitude-longitude spherical grid. In those studies, pieces of data are normalized by de-seasonalizing values which have significantly lower autocorrelation than raw data and detrending it by fitting linear regression model and retaining only residuals. Complex network is constructed by considering each spatial point in data set as vertex and statistical relationship between corresponding times series pairs (anomaly) [188] as weighted edges. At a global scale, number of nodes, density of edges, clustering coefficient, and characteristic path length were all examined and interpreted in context of climate. At regional scale, regions of high intensity are connected to a large fraction of globe, and hence can be interpreted as indicators for global climate system such as El Niño-Southern Oscillation (ENSO) or Pacific Index.

Pruning technique is used to avoid correlation computation for all possible pairs of nodes, its results are a fully connected network [170] since most edges have very

low weights. From that, properties of networks are analyzed to gain insights in climate domain. Clustering is performed on the network, then each cluster allows prediction of target variables with linear regression.

Clustering is also used to find interesting pattern in earth science data [168]. Common climate indices (time series of temperature and pressure in this study) are represented by performing it on the climate data with Singular Value Decomposition (SVD) and Shared Nearest Neighbor(SNN) techniques. Its applications for discovery of common indices on complex network can be considered as the conclusion for that work.

Building a complex network from climate data is also mentioned in [188], however, the study skips depersonalization step and focuses more on interpretation of complex network properties exhibited by the climate system. In the study, the network is composed by an architecture with two 'small world' networks: one operating in tropics and another acting as agent establishing links between the two hemispheres. That architecture provides extensions of other complex systems with many degrees of freedom and interactions at several space and time scales.

3.2.2 Data and simulation

Data from sensor networks

Sensing data from sensor stations provides information to enable humans and machines to understand better surrounding conditions [63]. Actually, those samples of data are categorized into three types: data generated by sensors, data describing sensors, and data describing environment [10]. Integration of those types of data is able to support designing, evaluating and planning public policies linked to the management of natural resources.

A collaborative approach for metadata management is introduced to acquire and manage metadata related to the physical devices and their surrounding environment [67]. The data aspect of this approach considers three types of data:

1. Sensor observation data. The actual measurements generated by the sensor. For example, humidity sensor measures and reports the relative humidity in the air.
2. Sensor metadata. The model which describes sensors and their capabilities in third parties format such as Sensor ML (Sensor Modeling Language) ¹ or the Semantic

¹SensorML provides a means by which sensor systems or processes can make themselves known and discoverable - <http://www.ogcnetwork.net/sensorml>.

Sensor Networks Ontology ².

3. Data processing rules. Specific rules which help to standardize management of data in applications and for decision support.

Metadata is "data which provides information about other data" ³. In other words, it contains necessary information to understand and use data. These information can be either of following types [127]:

- Descriptive metadata. It describes information for discovery or identification.
- Administrative metadata. It provides management information.
- Structural metadata. It depicts the container of information and indicates how data elements is put together.

Metadata is critical in multi sensor analysis.

Practical aspects of data integration

A framework, which allows combination of different data sources and analysis of the integrated data at different scales, is proposed in [183]. It provides means to handle big data from different data sources and perform analyses as well. Its principle is to combine Multi-Agent Simulation and Business Intelligence methodologies to adapt several purposes with among others: model and execute multi-agent simulation, manage input/output data of simulations, integrate data from different sources and analyze high volume of data. The framework focuses on simulation data to calibrate simulation as well as to manage a large amount of those sample data and to make analysis supporting decision-making processes. However, it seems that there is no information of sensor data and sensor description in the framework.

To maintain meteorological data, Meteo France ⁴ has a structure to include five datatypes [140] where each has abilities to access to the primary data as well as to understand its associative and descriptive realms. Actually, real data is stored in meteorological formats (normally BUFR ⁵), or in relational models, or in mixing of two methods. To

²an ontology which describes sensors - <https://www.w3.org/2005/Incubator/ssn/ssnx/ssn>

³<https://www.merriam-webster.com/dictionary/metadata>

⁴<http://www.meteofrance.com>

⁵Binary Universal Form for the Representation of meteorological data (BUFR) - a binary data format maintained by the World Meteorological Organization (http://www.wmo.int/pages/prog/www/WMOCodes/WMO306_vI2/LatestVERSION/).

retrieve it, one routine inspecting to the associative and descriptive parts to access the raw one in the primary structure. Therefore, each datatype represents metadata which allows retrieving data from the primary structure.

3.3 Cellular automata and sensing machine

3.3.1 Sensing station and sensor coverage

In figure 3.1, sensing devices monitor a physical parameter which can come from climate events or be an extreme phenomenon as insect invasion, flooding, rainfall. These equipments are deployed over a geographical region called R to measure parameters relating to climate. A collection of them called C composes a network which uses wireless connection or other communication types.

Each device is able to observe its vicinity. Sensing field of device s_i , referred as $f(s_i)$, is the area around it. The field $f(s_i)$, which is covered by device s_i , means that any point $\in f(s_i)$ is covered by s_i . If $f(s_i)$ is a disk, the sensing range r_{s_i} of $f(s_i)$ is the disk radius (figure 3.7). If the region R has an area $S(R)$, the minimum number of equipments to monitor the whole area is $N \geq \frac{S(R)}{\pi r_{s_i}^2}$.

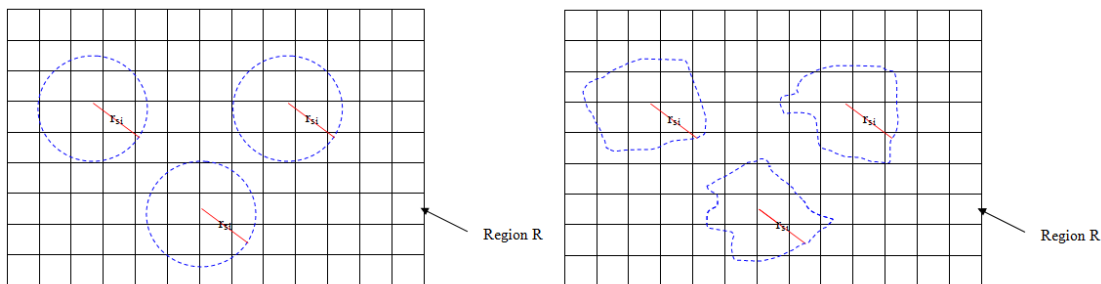


Figure 3.7: Coverage of sensing devices in a geographic area. On left ideal case homogeneous sensing range r_{s_i} , on right more realistic case complex sensing field $f(s_i)$.

The collection C gives a set of measurements of the region R. A question is what are current situation and trend of physical world? One way to answer is *to simulate the phenomena according to values given by sensing devices*. As a result, physical world model is necessary.

3.3.2 Coverage problem

One important problem highlight in figure 3.1 is *is the observation network qualify? Or in other words, how measurements of values are really reflecting the physical world?*. This issue relates in many aspects to sensor placement, phenomena nature or even understanding.

The observation simulation and physical simulation are coordinated to understand what is observed. By simulating a physical system it can be guessed what will happen at any places and any time. As a result, the sensor coverage can be optimized related to the physical evolution to monitor. For example, simulation of river flow allows to guess where and when to observe species or product dissemination.

In some monitoring applications, deployment of a large enough number of sensing devices is infeasible, as a result, small number of equipments are distributed in a region R . Issue here is to identify the minimum of stations to cover the region, where each device has the sensing field $f(s_i)$ with the range r_{s_i} . More clearly, it is about minimizing number of devices in respect of the k covering monitoring area [179][26]. For example, polynomial-time algorithms [75] are proposed to determine redundant stations so that they can check whether every point in service area of the network is covered by at least k stations.

In practice, coverage area of a network can be affected by surrounding conditions and terrain. An approach considers obstacles as interfere elements of sensing process and compose a shadow-fading model [187], model similar to radio wave propagation. Another is to assume that physical event arises in each grid point and it is detected with a probability defined between 0 and 1 (Elfes model [45]).

In addition, station positioning may depend on experts in domain fields and on other factors. For example, escape strategy and other management methods [130][71][32] are applied successfully to control insect outbreaks for rice in Vietnam. For these solutions, people use light traps to capture insects and the amount of insects becomes an indication to understand current situation and predict state for a wide area in next days.

3.3.3 Maps data and cellular automata modeling

Map is a *diagrammatic representation of an area of land or sea showing physical features, cities, roads, etc* (⁶). Map can help to locate a position on earth as well as

⁶Oxford dictionary

identify distribution of a mountain, a river or even populations.

The geographical region R, represented by a map where phenomena take place, can be considered as a cellular system by being divided into cells (figure 3.8). A cell consists of several attributes such as width, height, geographic position (including elevation). Cell is elementary unit of a cellular automaton, so it is indivisible.

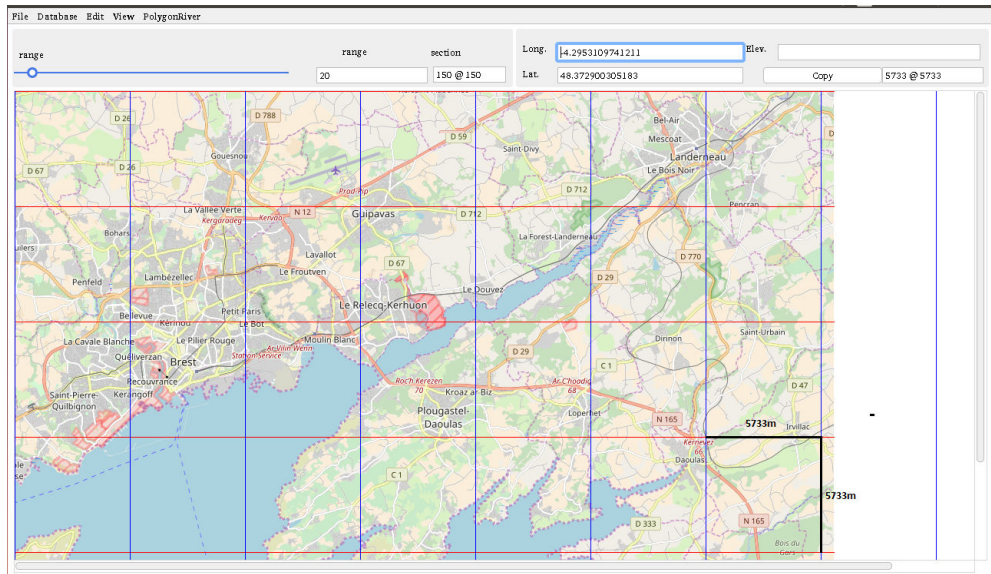


Figure 3.8: Map division into cells to compose a spatial cellular automaton structure, case of Brest harbor with 7 on 4 cells (grid size = 5733m x 5733m).

Sensor stations, which are distributed over the region R, can be considered in cellular automaton. Each station belongs to at most one cell, and vice versa, each cell may consist of at most one device.

A spatial cellular automaton described in section 3.3.3 is a kind of synchronous system where each cell represents a process and where link between a cell and its neighbors (Von Neumann, Moore, hexagon) compose edges allowing exchange of messages.

One view point is that CA synchronous system is a *sensing machine* where each cell is called as a process. Those processes execute in parallel the same program (MG and ST functions) to compose processes' new states (figure 3.9). These local executions respect a principle: a process has responsibility for one thing at a time. It is similar to how nature operates: local operations happening everywhere at once.

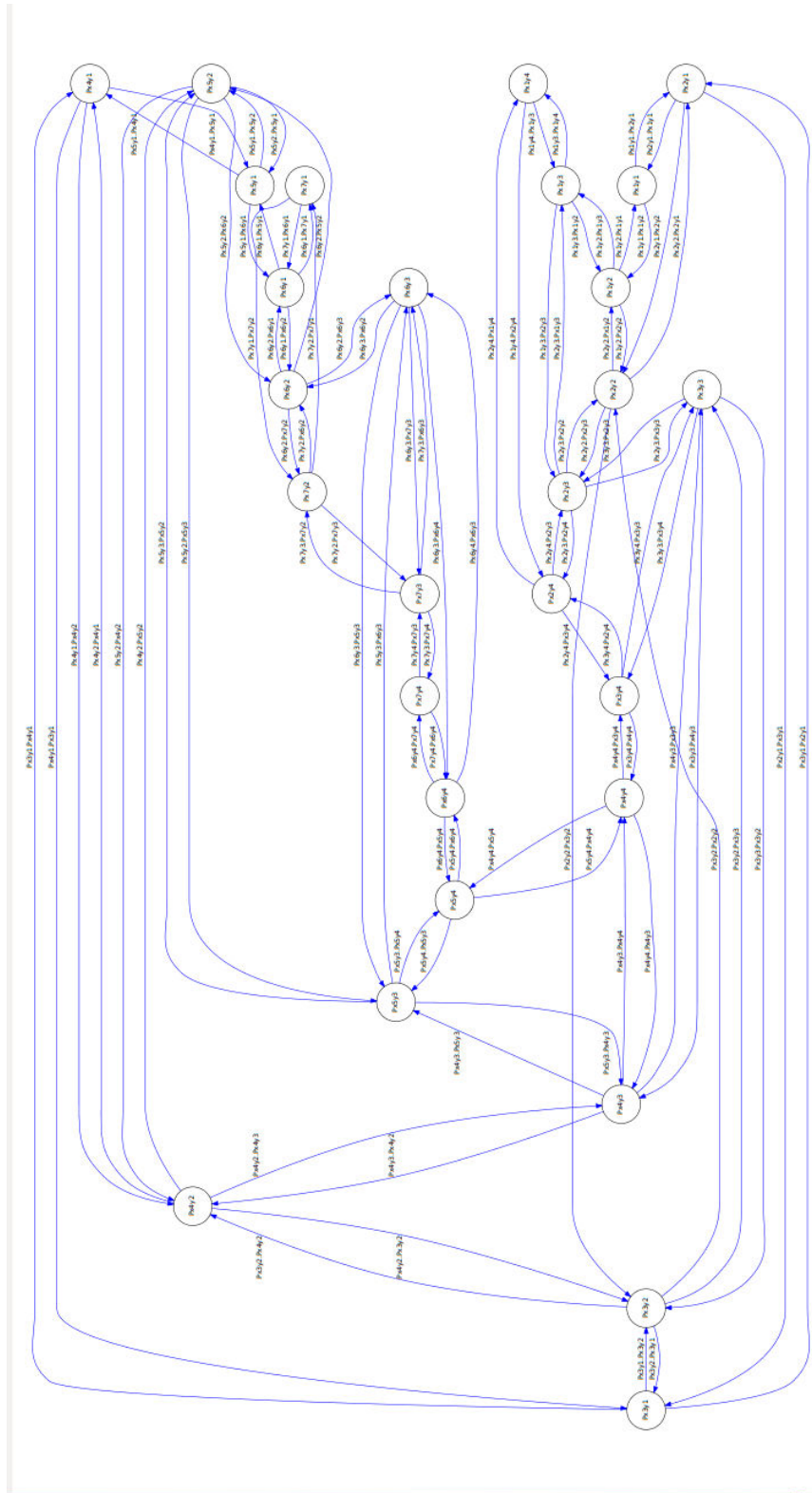


Figure 3.9: Sketch structure of a synchronous observation system related to CA: cells or process run same program at a time (parallelism and locality are described).

Conceptually, an environment sensing machine consists of several overlapped cellular automata (figure 3.10) in which the spatial one, referred as the host cellular automaton, represents the geographical region R . Because each state of a process is a composition of values including meteorological and other factors, for all processes over the host one, each factor creates another cellular automaton. The compound CA system at a time t_i is called a configuration of the sensing machine at that time.

Temporal aspect of the machine is also described in figure 3.10. In fact, each process of the host cellular automaton is positioned as (i, j) where i, j are row and column indices. A transition rule method $f_t(i, j)$ (including MG and ST functions), applied synchronously on all cells at the time t , specifies a new configuration to be computed from states of neighbors of (i, j) . Cell (i, j) itself can be considered as an adjacent as well.

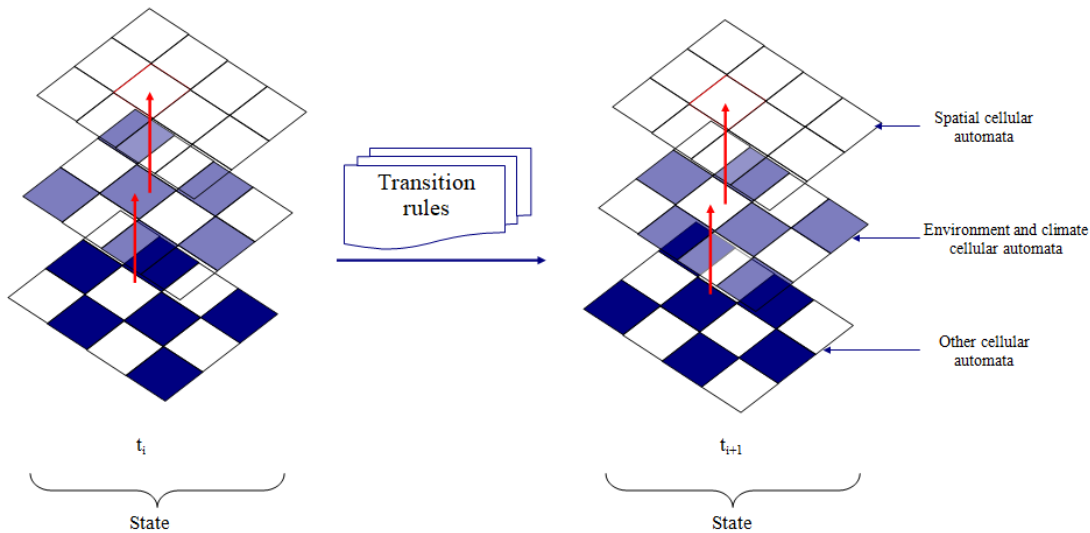


Figure 3.10: Structure of a complex multi-layer cellular automata machine at two consecutive states (t and $t+1$): transition rules (MG and ST functions) described the state change.

3.3.4 Coverage issue modeling

The problem presented in section 3.3.2 can be addressed and modeled using CA. Indeed, the map representing the monitoring area is divided as cells while each sensing radius r_{s_i} depicts a phenomenon to model (figure 3.7). Thus, shadows-fading [187], Elfes [45]), or other methods can be implemented by distribution method using automata machines.

In the implementation, parameters which interfere with behaviors of sensors can be populated from external data sources. These elements including building, meteorological factors, terrain will be discussed in next sections as a data aspect of CA modeling.

3.4 Data access architecture

3.4.1 How practical data is accessed

Architecture in figure 3.11 presents different ways to access to data in environmental monitoring systems. Several databases are stored in a GIS **Relational Database Management System** where there are metadata which illustrates how data is organized in storing databases. Usage of GIS enables maintenance of spatial data such as: river, building, road. In implementation level, OpenStreetMap (OSM)⁷ is used and maintained in Postgres SQL.

Before storing external data to GIS RDBMS, some preprocessing steps are done to normalize raw data from different sources and formats to adapt with the data organization. External users are able to access data via different ways such as: web, CSV, SensorML or even MetView⁸. In addition, internal users can make queries to metadata as well as storing databases to elaborate future tasks.

A metadata database (description database or datatype) is the core of the architecture (figure 3.11) to provide descriptions about the maintenance of storing data. For example, what data types of insects are kept in insect monitoring application? Where are environmental factors coming in this application? Or which sensors are used as well as what are their types of data values?

The most important role of metadata is in data reuse and data sharing. Highly detailed instructions or documentations of data are supported in order to interpret and analyze data accurately.

In environmental monitoring applications, metadata is more necessary since a lot of information involving to sensors need to be maintained. These kinds of information relating to historical calibration and coefficient data of sensors. Sensor values and types need to be described to better understand the surrounding conditions that sensors measure. In addition, in some cases, calibration characteristics of each sensor are stored before the

⁷<https://www.openstreetmap.org/>

⁸A meteorological workstation application designed to be a complete working environment for both the operational and research meteorologist - <https://software.ecmwf.int/wiki/display/METV/Metview>

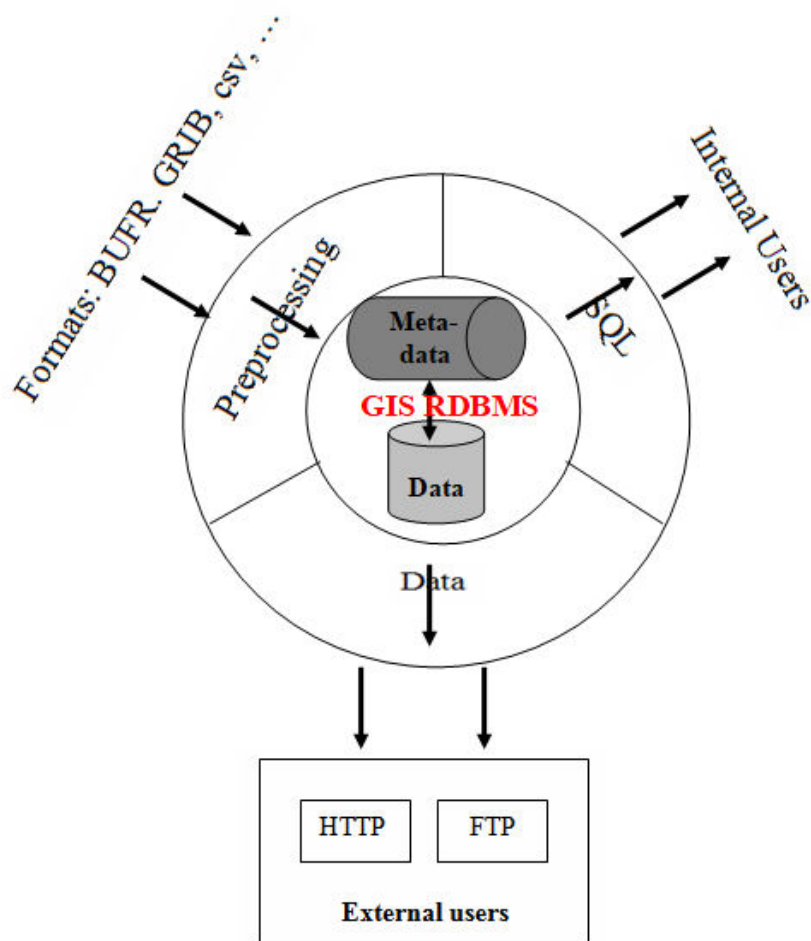


Figure 3.11: Architecture of data access in environmental monitoring systems.

sensor is deployed to a remote station.

For example, to monitor insects in rice fields, people deploy sensor stations (figure 3.12) to measure factors influencing insect growth such as temperature, wind, humidity, rain, insect density. These factors have different types of values such as floating point number (wind velocity, insect density), integer (wind direction, rain flow). Moreover, cameras which are used to count insect densities are calibrated to estimate their internal and external parameters such as image center, distorted parameters. All of those things need to be specified as metadata so that they can be stored for later purposes.

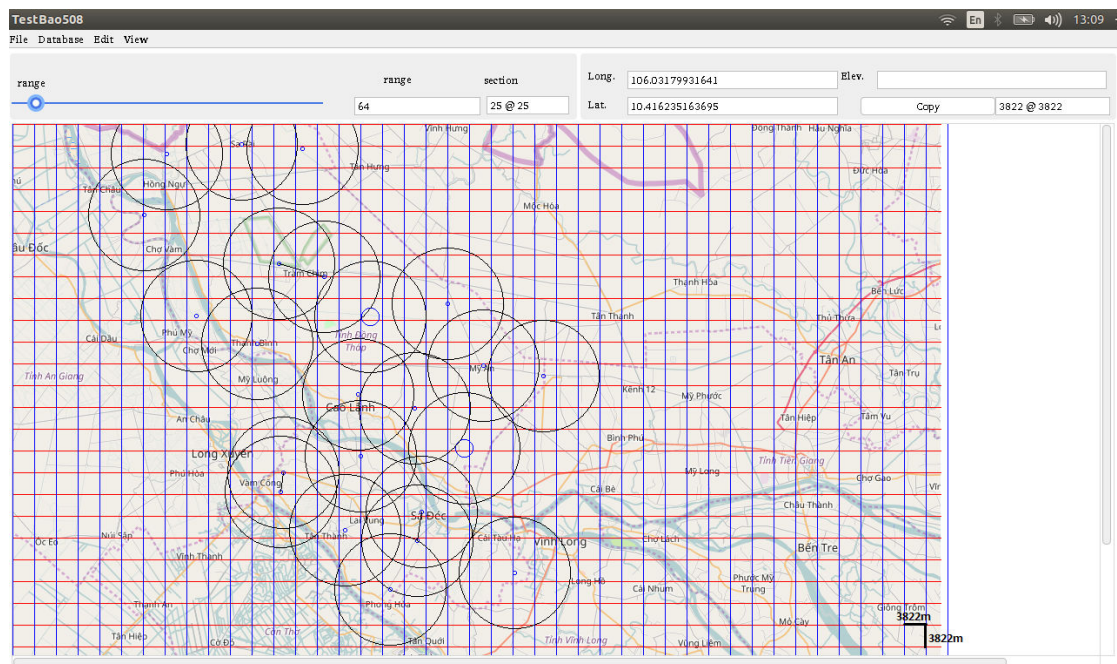


Figure 3.12: Representation of sensor stations distributed on study area to sample physical parameters (map with grid size = 3822m x 3822m).

3.4.2 Description of data in cellular automata sensing machine

Data in environmental monitoring systems (figure 3.13) can be populated from different sources such cellular automata (CA), meteorological, simulation objects, geographic and sensor data. The cellular automata data is the center in which all others are distributed into cells of the CA. It can be said that there are several layers of CA, which each represents from meteorological and geographical data to simulation objects or sensor data. These layers overlap together or are mapped to the lattice representation of the

spatial host cellular automaton (figure 3.14).

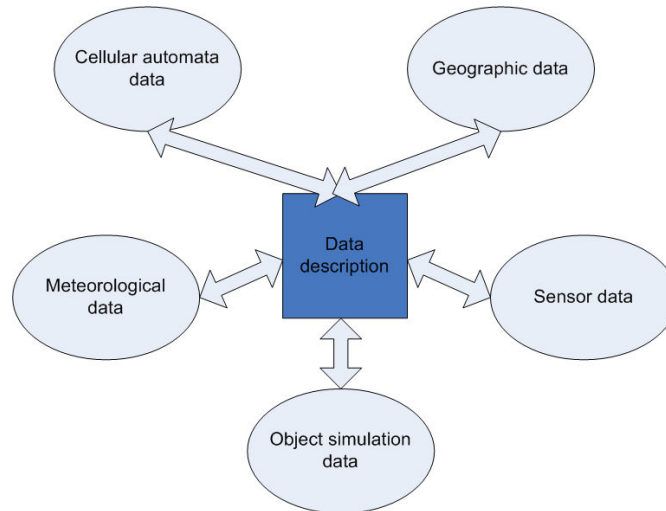


Figure 3.13: Interaction of data in environmental monitoring systems.

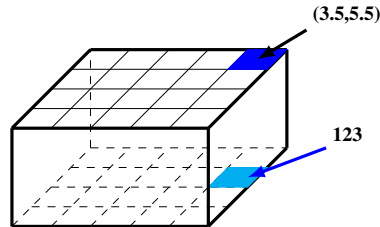


Figure 3.14: Two CA layers overlapped together, first layer stores wind vector (u, v) data in each cell and second one is holding insect density.

External data sources

Indeed, natural phenomena can accept inputs from various parameters. For example, insect invasions in Mekong Delta are influenced by wind, temperature, humidity, insecticide spraying, and others factors. Or urban flood inundation is affected by elevation, rainfall, the water depth, infiltration rate to previous surface, roughness coefficient and drainage rate of inlets [30][113]. Those kinds of data share the similar thing: they are provided by external data sources in different formats such as PostGIS, CSV files, GRIB

⁹, or NetCDF ¹⁰.

Based on time dimension, those kinds of data is divided into two categories:

- Less sensitive to time variation such as building, roads, rivers and other geographical features. They can be maintained in databases such as Open Street Map ¹¹ in PostGIS format.
- Continuous and frequently varying over time such as environmental parameters (temperature, humidity, ...) , universe (galaxies, planets, stars, satellites). They are stored in different self-describing standard files.

Meteorological data description External sources include weather data from meteorological service providers as well. For example, Météo-France maintains following meteorological datatypes [140] in their architectures:

- Point forecasts (surface and upper air) produced by models and forecasters.
- Quality/Control flags put on observations by models and forecasters.
- Observations made by either direct/indirect instruments or eyes of human observers.
- Numeric Models Outputs.
- Image.

However, that study focuses on forecast data from service providers under conditions of a spatial and temporal resolutions. Spatial resolution depicts how many forecasts within a given area while temporal one illustrates how often the forecast is updated. It means that a grid of points, which is updated periodically, are provided as inputs of preprocessing step in figure 3.13. For instance, wind and temperature data in figure 3.4 are retrieved from zyGrib (<http://www.zygrib.org> which provides services to retrieve weather data from National Oceanic and Atmospheric Administration - NOAA) under GRIB format with (0.5°latitude x 0.5°longitude) resolution.

⁹ (GRIdded Binary or General Regularly-distributed Information in Binary form) is a concise data format commonly used in meteorology to store historical and forecast weather data <https://www.wmo.int>

¹⁰ a set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data <https://www.unidata.ucar.edu/software/netcdf>

¹¹<https://www.openstreetmap.org/>

Object description

Monitorable objects, which are related to a phenomenon as either main actor or with supporting role, are described. For instance, in the case study, BPH is the main actor which causes pest outbreaks but rice is also necessary because it is the main food for hoppers.

Sensor station description

Sensor stations are geo-localized to measure the environment. Each station has its own sensors that hold values. These values can be seen as inputs of the phenomenon or a capable way to validate the model of the phenomenon.

Schema of data descriptions in EM applications

Schema in figure 3.15 depicts all above data descriptions in EM applications. In other words, it presents metadata of environment machines.

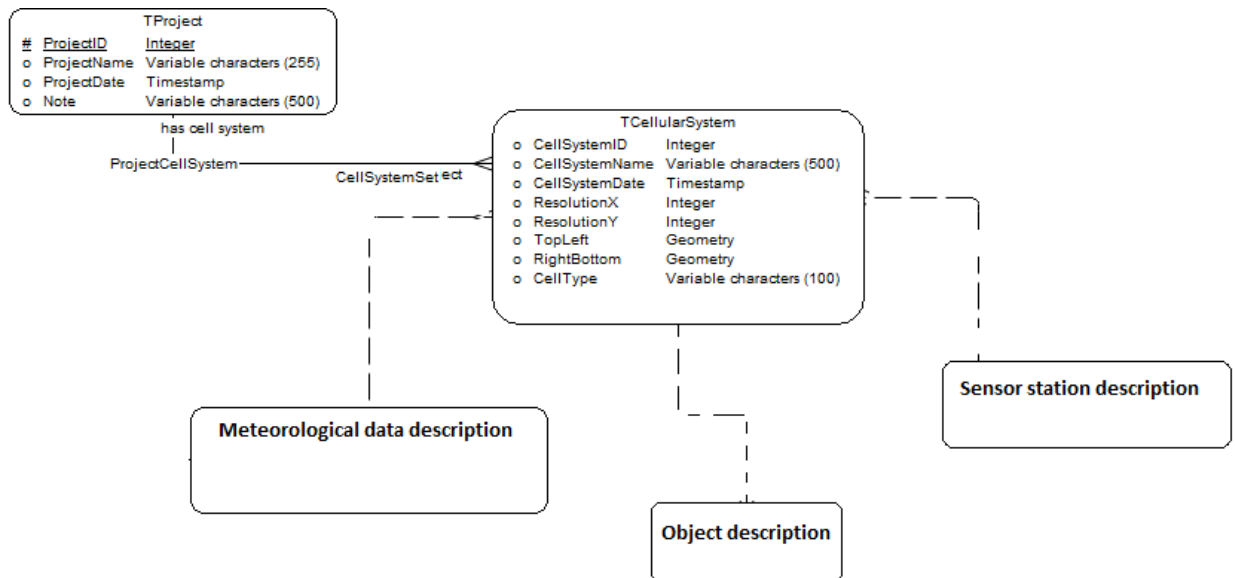


Figure 3.15: Description of data and object interactions used in environment machines, composed for meteorological data, object and sensor (more details in appendix D.1).

An application is termed as a project which has its own attributes such as: ID, name, creation date time. For example, table 3.2 depicts attribute values of 2 applications:

insect monitoring in Mekong Delta and radio signal propagation.

Table 3.2: Example of attributes of 2 applications: insect monitoring and radio propagation.

| ID | Name | Datetime | Note |
|-----------|-----------------------|---------------------|-----------------------------------|
| 001 | BPH Mekong | 2017-08-01 04:05:06 | Insect monitoring in Mekong Delta |
| 002 | Radio wave simulation | 2017-09-02 05:06:07 | Simulation of radio propagation |

Spatial CA descriptions is a main element of an EM application. An application (project) has at least 1 spatial CA. Main attributes of a CA are resolutions (or cell resolutions), top left and right bottom geo-locations of the region that phenomena take place. The CA description has relationships with descriptions of weather data, object, and sensor.

- Meteorological data description. It depicts weather factors used as well as their data types, data sources.
- Object description. It illustrates meta data of objects which participate to the phenomena. These descriptions include main attributes such as object name, object type.
- Sensor station description. It shows descriptions in details about stations as well as their sensors in the EM application.

3.4.3 Transformation to data definition

From the description data model, it is possible to generate a EM database structure by identifying some following information:

- Meteorological factors which impact phenomena need to be monitored. Descriptions of these factors such as data types, source are also provided.
- Objects representing phenomena. For example, in a flooding monitoring project, objects participating to the phenomenon (water level, rain fall, ground water) are depicted as well.
- Description of sensor stations (if any).

Table 3.3: Information of weather factors in an insect monitoring application.

| No | Meteorological factor | Domain | Length | Number of Precision |
|----|-----------------------|--------|--------|---------------------|
| 1 | Temperture | Number | 14 | 8 |
| 2 | Wind u | Number | 14 | 8 |
| 3 | Wind v | Number | 14 | 8 |
| 4 | Humidity | Number | 14 | 8 |
| 5 | Precipitation | Number | 14 | 8 |
| 6 | Pressure | Number | 14 | 8 |

Table 3.4: BPHs density and rice ages in insect invasion monitoring application.

| No | Object name | Attribute | Domain | Length | Precision | Number of items |
|----|-------------|-----------|---------|--------|-----------|-----------------|
| 1 | BPH | Density | Number | 14 | 8 | 28 |
| 2 | Rice | Age | Integer | | | |

All these above information are maintained in a database which is the implementation for the schema shown in figure 3.15. Next, a storing database is created automatically to allow data manipulation in the monitoring application.

For instance, the invasion of insects in Mekong (BPH) is influenced by wind, temperature, humidity, rain falls, pressure. Moreover, rice age is another factor which has ability to susten BPHs because young rice is their best food. Besides, it is necessary to estimate the number of BPHs in a region (e.g cell), therefore, a structure to maintain hopper densities is depicted as well. Those information are illustrated in table 3.3 and 3.4.

As a consequence, generated database has following layout (figure 3.16):

From the metadata schema in figure 3.15, a workflow for development an environmental monitoring application is emerged. It can be found in the appendix D.2.

3.5 Expected behaviors of the environmental machine

Behaviors depict the temporal aspect of environment machines (figure 3.10). In fact, a compound value is assigned to each process at *the time t* to compose the state of the cell at *the time t*. Because the value is a combination of simple ones, in an abstraction view, there are several cellular automata overlapped together (section 3.4.2).

In implementation, the machine consists of transition rules functions to depict such behaviors. Actually, those functions are represented as varieties of growth and propagation algorithms of CA.

3.5.1 Growth and propagation algorithms

Growth algorithm

When the new state $f_{t+1}(i, j)$ of a cell is calculated from its local state $f_t(i, j)$ (instead of all its neighbors' one), this type of rules is called a local rule. For instance, the rule 'Death rate of insects is 10% per day' is local since it is applied to the cell that contains insects. Or 'every day an adult insect may give 10 eggs' belongs to this type as well. All local rules in a cell compose the growth algorithm of this cell.

Propagation algorithm

The new state $f_{t+1}(i, j)$ of a cell is calculated from its neighbors. For example, the rule 'insect may transmit to adjacent area due to wind' composes a propagation algorithm for insects.

To implement the growth and propagation algorithms, bilinear interpolation is necessary since sensing devices provide data at sampling locations, however, values in all over the spatial cellular automaton are required. In addition, in the beginning, it is required to downscale gridded meteorological data to the grid of the host cell system because these grids are often different in terms of resolution.

3.5.2 Behavior implementation

The concurrent model of cellular automata synchronous system requires usage of a parallel programming language in the implementation phase. The tool PickCell [15][115] can produce CA codes in destination programming languages. The integration of features described in section 3.4 contributes to the code generation as well. Occam and CUDA are currently used in this phase.

Occam language

Occam is a parallel programming language developed by David May [117], based on Communicating Sequential Processes (CSP) [72]. It is used for implementing synchronous system in CA machine and synthesis, message behaviors for WSN.

In Occam, cells are processes and are able to be executed concurrently. Passing messages along point to point channels is used to communicate among those concurrent processes [40]. Figure 3.17 depicts a message sending from process P1 to process P2 along channel C.

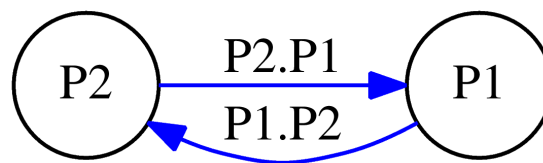


Figure 3.17: Description of message sending in cellular automata network: process P1 sends a message via channel C to process P2 (from [40]). It is generated from the tool PickCell.

CUDA language

CUDA (Compute Unified Device Architecture) is a parallel computing platform which allows developers use CUDA-enabled graphics processing unit (GPU) for general purpose processing (GPGPU) [24].

A kernel, parallel portion of a CUDA application, consists an array of threads which execute the same code. Therefore, for realization synchronous system, cells in CA are depicted as threads in CUDA kernel and their transition rules functions are described as kernel codes. Figure 3.18 depicts how a CUDA application executes by transmission data from CPU from/to GPU and processing in GPU.

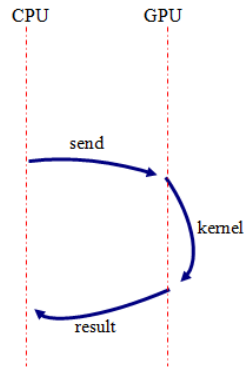


Figure 3.18: Schema showing how the execution of a CUDA application. The CPU sends data to GPU to manipulate and receives data results after processing for later purposes.

3.5.3 Some algorithms in spatial CA

Bilinear interpolation for meteorological data

Figure 3.1 depicts the distribution of sensor stations in some locations. Thus, to calculate meteorological factors for all points of CA grid, interpolation is expected to be run.

Bilinear interpolation [142] (figure 3.19) interpolates the value of I ($f(I)$) based on values of its 4 neighbors $A_1(x_1, y_1)$, $A_2(x_1, y_2)$, $A_3(x_2, y_1)$, $A_4(x_2, y_2)$:

$$f(I) = \frac{(x_2-x)(y_2-y)}{(x_2-x_1)(y_2-y_1)}f(A_1) + \frac{(x-x_1)(y_2-y)}{(x_2-x_1)(y_2-y_1)}f(A_2) + \frac{(x_2-x)(y-y_1)}{(x_2-x_1)(y_2-y_1)}f(A_3) + \frac{(x-x_1)(y-y_1)}{(x_2-x_1)(y_2-y_1)}f(A_4) \quad (3.1)$$

Because meteorological values are measured at defined points on the earth (spherical surface), when formula 3.1 is used to interpolate meteorological values, subtraction operators are replaced by distance between two points on the earth's surface.

Algorithm in appendix B.2 depict an implementation in CUDA. It uses the structure 3.1 which is generated automatically from PickCell for CA cell declarations.

```
typedef struct CellPosition {
    int identity; // identity of the node
    int x,y; /* cell geometric position */
    double longitude , latitude , elevation; /* geo position */
}
```

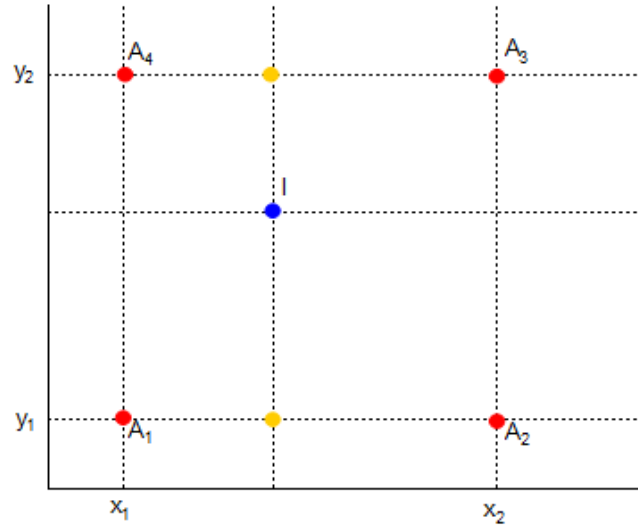


Figure 3.19: Description of bilinear interpolation at point I based on its four neighbors A_1, A_2, A_3, A_4 .

```
} CellPosition;
```

Listing 3.1: Geometric and geographic declarations of cells generated from PickCell

Distance between points on the earth's surface

Haversine formula [162] determines the distance between two points on a sphere given their longitudes and latitudes.

In figure 3.20, let α_1, ϕ_1 and α_2, ϕ_2 be the geographical latitude and longitude in radians of two points P and Q. $\Delta\phi, \Delta\alpha$ are their differences and $\Delta\sigma$ is the central angle between them. Then the distance d between P and Q, the arc length, is given:

$$d = r\Delta\sigma \quad (3.2)$$

Where r is the radius of the sphere and $\Delta\sigma$ given in radians.

$\Delta\sigma$ is calculated as followed:

$$\Delta\sigma = \arctan \frac{\sqrt{(\cos\phi_2 \sin(\Delta\lambda))^2 + (\cos\phi_1 \sin\phi_2 - \sin\phi_1 \cos\phi_2 \cos(\Delta\lambda))^2}}{\sin\phi_1 \sin\phi_2 + \cos\phi_1 \cos\phi_2 \cos(\Delta\lambda)} \quad (3.3)$$

CUDA implementation of this algorithm can be found in appendix B.2.

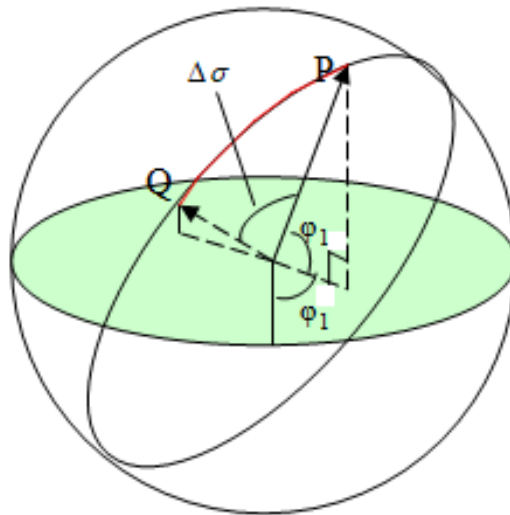


Figure 3.20: Distance between two points P, Q on a sphere of center O. α_1, ϕ_1 are the longitudinal and latitudinal angles of P, respectively. $\Delta\sigma$ is the angle between the two segments OP and OQ.

Wind calculation

A horizontal wind vector w is represented by two components (u, v) (figure 3.21), where:

- Zonal velocity u . The component of the horizontal wind TOWARDS EAST.
- Meridional velocity v . The component of the horizontal wind TOWARDS NORTH.

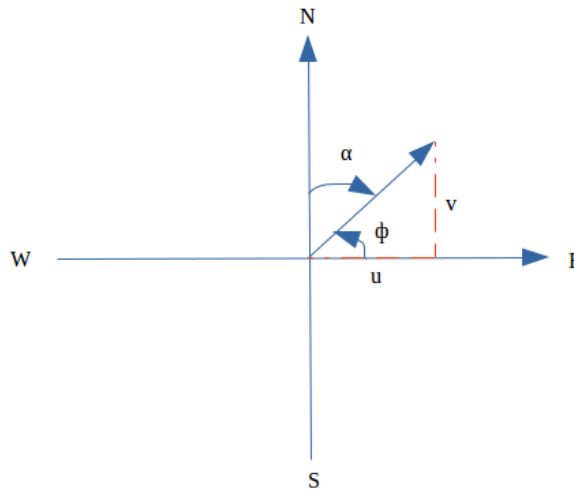


Figure 3.21: Velocity wind vector components u and v in meteorological wind direction.

Let α be the direction TOWARDS which the wind is blowing and ϕ be the wind vector polar angle in two-dimensions. The wind velocity and the polar angle are calculated as:

$$\begin{aligned} |w| &= \sqrt{u^2 + v^2} \\ \phi &= \frac{180}{\pi} * \arctan 2(u, v) \end{aligned} \quad (3.4)$$

3.6 Data querying in environmental monitoring applications

Executions of EM applications produce data which can be used for post investigations such as decision making, reuse or representation. Some questions may be raised in that matter. For instance, what is the tendency of insect invasion What is wind direction in an area in past two years? Can a monitoring application represent monthly growth chart of

insects? These kinds of problems can be answered by querying the generating database (as the example in figure 3.16) and the description database (figure 3.15).

This section shows 2 examples of querying data from an EM application. The first example is about wind rose calculation at a location after wind vector evolves after a number of simulation steps. It means that after each step, simulation results are populated to the database (as schema in figure 3.16). When the simulation finishes, an algorithm under SQL query is called to compute the wind rose at a location. The second example shows how to retrieve sensor data from the database in schema 3.16).

3.6.1 Wind rose algorithm

*A wind rose gives a succinct view of how wind speed and direction are typically distributed at a particular location*¹². Wind roses were predecessors of the compass rose [28] and by representing on a polar coordinate system, the frequency of winds over a time period is plotted by wind direction, with color bands showing wind speed ranges.

From the database of which schema shown in figure 3.16, the wind rose dataset for a location in the application (Mekong Delta in this example) can be created by querying data in table `TMeteoToBPHData` with calculations using formulas 3.4.

In the application *Insect Mekong*, table `TMeteoToBPHData` contains 164388 records. These records are populated after each step of the insect invasion simulation (in CUDA). The CUDA implementation allows querying data as algorithm 1, grouping data and displaying the wind rose as following figure 3.22.

Figure 3.22 depicts the wind rose captured after executing insect invasion simulation in a Cao Lanh, Vietnam. It can be seen that the direction of wind tends to come from South West (SW), SSW and S. This tendency is similar to the one of the data in the beginning of September, 2017 (www.zygrib.org/).

3.6.2 Data collection from sensor stations

Data collections from a sensor station, sometimes, need to be extracted to better understand situation at the station's location. Some questions about the information system may arise around stations and their values. For example, what are sensor values of a station at a period of time? What is the tendency of BPHs density at a sensor station in the past three months? What are tracks of a node if these devices are mobile?

¹²<https://www.wcc.nrcs.usda.gov/climate/windrose.html>

Algorithm 1: Query to calculate wind velocity and direction.

Input: LocationID of cell, Table TCellularSystem, TCell, TMeteoToBPHData

Output: Wind velocity, wind direction (angle) of a point in the application 'Insect Mekong'

```
SELECT  windu, windv, sqrt(windu*windu+windv*windv)/5 as mg,
        CASE  WHEN (sqrt(windu*windu+windv*windv)/5 < 5) THEN '<5km/s'
        WHEN (sqrt(windu*windu+windv*windv)/5 < 10) THEN 'From 5 To <10km/s'
        WHEN (sqrt(windu*windu+windv*windv)/5 < 15) THEN 'From 10 To <15km/s'
        WHEN (sqrt(windu*windu+windv*windv)/5 < 20) THEN 'From 15 To <20km/s'
        ELSE '>20km/s'
        END WindGroup,
        round((cast((270-atan2(windv,windu)*180/3.1457) as int)%360)/22.5)
        as angle
FROM    tcellularsystem cs Inner Join TCell c
        On  cs.cellsystemid = c.cellsystemid
        Inner Join tmeteotobphdata m On c.cellid= m.cellid;
WHERE  cellsystemname = 'Insect Mekong'
And c.CellID = LocationID; -- parameter
```

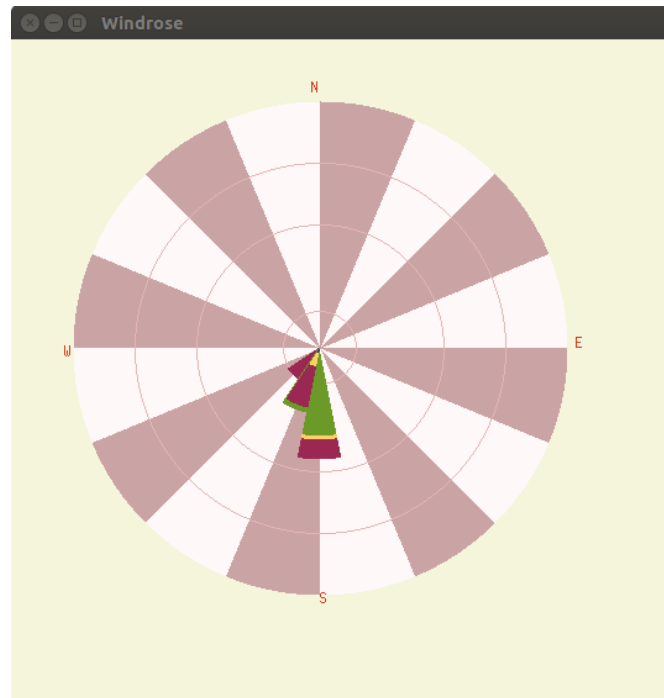


Figure 3.22: Wind rose computed at a point of insect invasion application after executing the simulation.

The following algorithm (algorithm 2) illustrates how to retrieve sensor values from a station in an EM application. Implementation of the query is done by a function in Postgres SQL; next a Smalltalk program is able to call and represent it as figure 3.23.

3.7 Summary

The chapter advocates a method to model physical world as CA machine. The machine consists of cells (processes) which are able to execute synchronously same transition rule method to compose a new configuration. In addition, sensing stations are distributed over its geographical region to monitor a service area (coverage) of the physical world.

In addition, the chapter depicts an information system to contribute to the development of global sensing machine in modeling environmental monitoring. Technically, the system allows integration of data from several sources such as: observation points, meteorological services, GIS. Its structure is maintained in PostGIS to take advantages of GIS data (points, polygons, shape data) and to integrate other data with the assistance of a proper database description. Besides, a method combining CA and data approach is

Algorithm 2: Query to get data from a sensor station in the application.**Input:** Table TBPHSensorValues**Output:** Temperature, Wind velocity, wind direction (angle), Humidity, BPHs density, Not BPHs density on 10/09/2017 at station 10

```

SELECT  temperature, windu, windv, sqrt(windu*windu+windv*windv)/5 as mg,
        round((cast((270-atan2(windv,windu)*180/3.1457) as int)%360)/22.5)
        as angle, humidity, BPHDensity, NotBPHsDensity
FROM    TBPHSensorValues
WHERE   Cast(NodeTime as Date) = '2017-09-10' -- ISO-8601 format, YYYY-MM-DD
And     NodeID = 10;

```

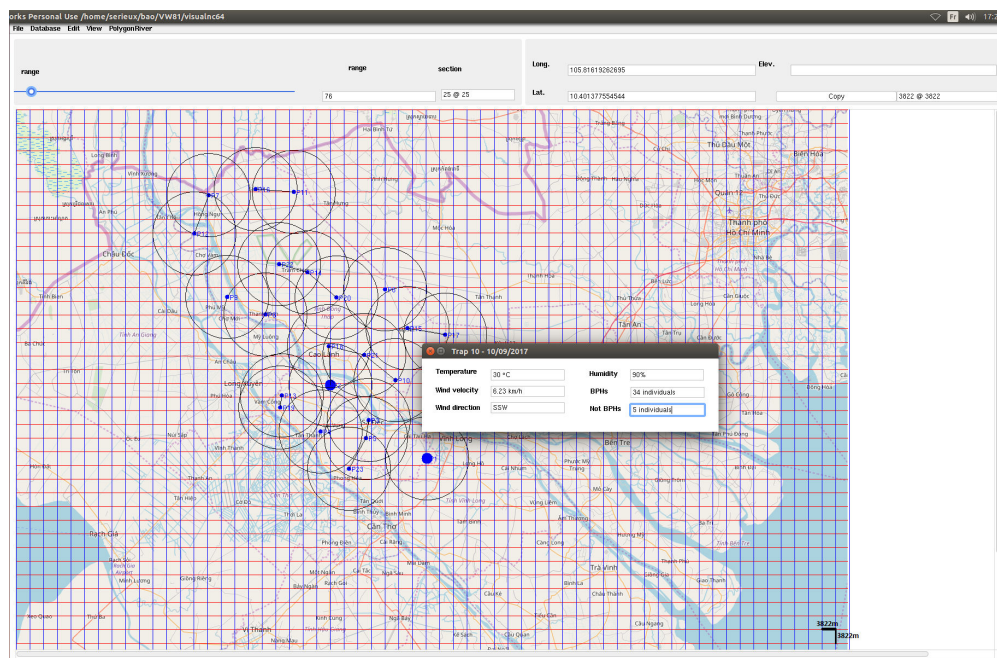


Figure 3.23: Example of accessing to values of the sensing station 10 from map, values include temperature, wind, humidity, BPHs and not BPHs densities (Map with grid size = 3822m x 3822m).

also introduced to simulate natural phenomena and extract data for post processing.

The information system provides means of data in order to develop third application such as visualization as in <http://www.smartmekong.vn/> and sames.univ-brest.fr. By the time passing, when the volume of data starts to be large, business intelligence (BI) solutions may take advantages from it for analyzing and presentation.

4

Application: Modeling rice insect pest using cellular automata machine

4.1 Introduction

Insect pests can cause severe crop damages. For example, locust pests create dense insect clouds and destroy every plants they found by eating them. In March 2013, nearly 50% agricultural surface areas of Madagascar have suffered from swarms of locusts, with billions of individuals in each swarm [1]. This infestation devastates 60% of rice crop of the country and costs millions of dollars for pest control. Another example is Brown Planthoppers (BPHs) which cause hopper fire in young rice and damage several hundred thousands tons of rice production in Vietnam.

Several solutions to confront with insect pests are proposed and solutions with sensor networks emerge as suitable choices [109][185]. This kind of solution uses sensors to

measure environments and insects behaviors. Sensed values will be sent via a network to a data center. Next, a back-end system will manipulate these values and propose solutions relating to data collection. Such application is a kind of environment machines shown in figure 3.1.

To implement such system, it is important to understand factors influencing insect behaviors and use them as model. Actually, these factors occur continuously and concurrently. Continuous occurrence means that they compose an unbroken whole while concurrency allows them to happen at the same time. For example, some factors such as temperature and wind are influencing BPHs invasion [125][137] from one place to another, in a continual way. Besides, they are concurrent since motivation for propagating from a source to a destination comes with surrounding conditions of the source and its neighbors. Conditions of different places must be executed simultaneously.

In this chapter, we apply the sensing machine based on synchronous system [114] and cellular automata [206] topology to depict the concurrency and continuity of influence factors. In fact, this chapter is an illustration of local and global observation systems which are mentioned in chapter 2 and chapter 3.

The rest of this chapter is as follows. Section 2 is about modeling insect invasion phenomenon as a synchronous system with cellular automata topology. Next section depicts some algorithms to implement the model described in section 2. Section 4 illustrates simulation results of the insect invasion monitoring application. The last section is a summary in respect with cyber physical aspects developed in chapter 2 and chapter 3.

4.2 Modeling insect invasion using cellular sensing machine

4.2.1 Cellular automata synchronous system

The map representing the working space of insects is divided in cells. In other words, the map is modeled as a lattice of a cellular automaton (figure 4.1).

The space in figure 4.1 depicts a spatial cellular automaton which is described by a triple $(\mathbf{S}, \mathbf{n}, \mathbf{f})$ where:

1. \mathbf{S} is a finite set of state. Each cell has a state, composition of environmental factors

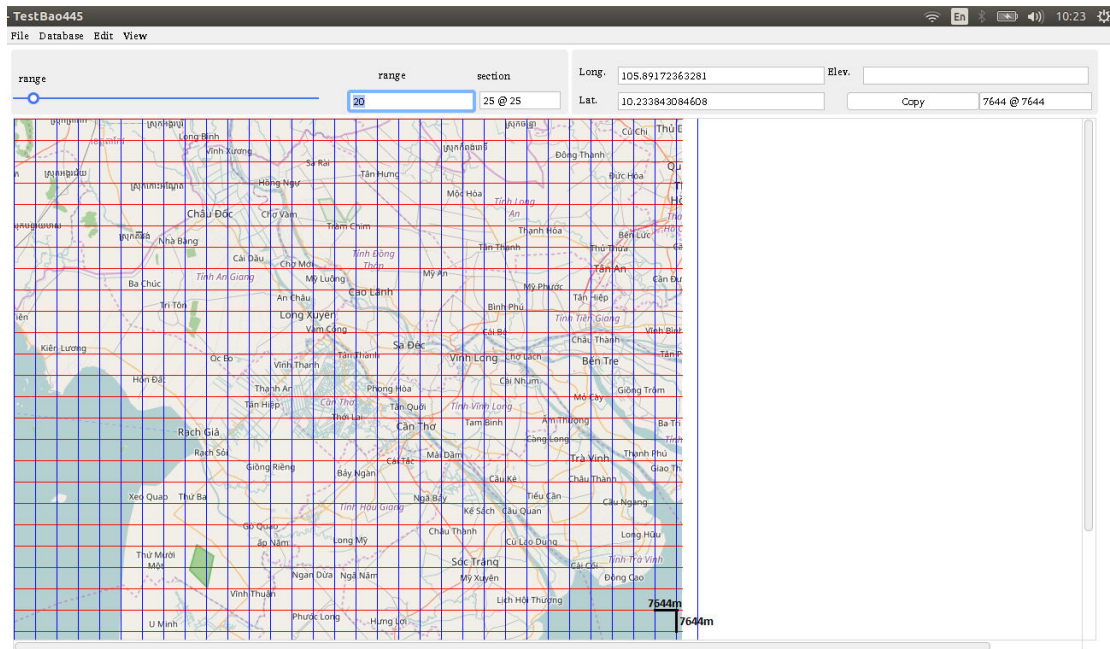


Figure 4.1: Map representing insect behaviors in their working space divided in as cells (grid size = 7644m x 7644m).

and biological of monitorable objects in the insect pest, to illustrate its status.

2. Distance n identifying neighbor cells, normally $n=1$. When $n=1$, a cell has at most 4, 6, or 8 surrounding cells.
3. Transition rule $f: S^n \rightarrow S$ depicts the change of a cell's state at a specific time based on its current state neighbors' values.

In figure 4.1, the space is represented by a matrix of cells, 2-dimensions. However, in theory, this space has an infinite number of dimensions. Each cell in the space can, at any given time, be in a finite number of states. At the time t , state of a cell depends on state at time $t-1$ of its neighbors. The cell itself can be integrated in its neighborhood. Updating the rules is identical to all cells. Whenever the rules are applied to the entire system, they could change the entire system synchronously.

CA are capable to combine spatial and temporal relationship (figure 3.10). Lattices, cells, neighborhoods depict spatial characteristics. In fact, each cell represents a variable that hold the cell's values at the time t . Transition rule f manages changes in variables of the cell and its neighbors from time t to time $t+1$, given as input automaton outside data of time t .

CA also imply data aspect. Indeed, values of cell's variables may come from either environmental or biological factors. For example, in flooding modeling, water level, downstream or upstream are an important factors to identify where water moves to. Geology and river floor at a cell are considered as well.

As a result, CA modeling is constituted of three aspects: spatial, temporal and data (figure 4.2).

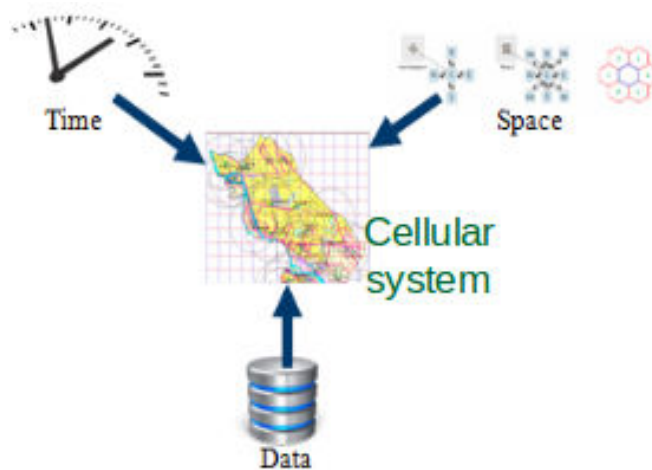


Figure 4.2: Description of CA modeling interactions.

4.2.2 State variables of cells

The space where insects live and act is divided as cells shown in figure 4.1. It is considered as a graph G of which nodes are cells and edges are created between a cell to its neighbors.

By the time passing, each cell updates its values with the assistance of a function f . The function f depends on the cell's states which is composed of either environmental or biological factors: wind, temperature, humidity, density of hoppers, rice ages. These elements come from external data sources, monitorable objects or sensor stations as described in section 3.4.2.

Meteorological data

In short, a state contains following factors:

- Temperature and humidity. Because they have influence on all phases of hoppers.
- Precipitation. There is a relation between rainfall and hopper death rate.
- Wind: the wind magnitude. According to biological characteristics of hoppers, most part of the propagation (only adult ones) are due to wind advection, thus strong wind help the propagation.

Biological data

The most important biological factor is the hopper itself (hopper life cycle). Because life cycle is around 28 days and is spread in 3 phases, it can be considered as an one dimensional cellular automaton of which cell i contains the number of i days old BPHs (figure 4.3).

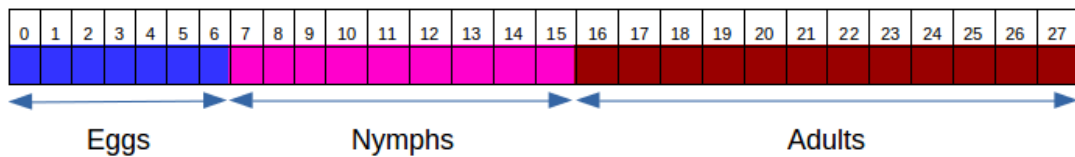


Figure 4.3: Biological cellular automaton illustrating life cycle of BPH in cell based on its age in days.

Let the number of i days old BPHs is $\mathbf{n(i)}$, after a day, principle of updating cell i (without weather conditions) in the biological cellular automaton is $n(i) = n(i - 1), \forall i \in [1, 27]$.

In addition, rice is an important factor which influences insect behaviors because insects consider young rice as their main food.

To sum up, variable of a cell consists of two parts:

- Weather data including: wind, temperature, humidity, precipitation.
- Biological data. The biological cellular automaton is depicting the number of hoppers which are i days old $\forall i \in [0, 27]$. In addition, rice age is another input parameter.

Sensor stations

Some sensor stations are deployed to measure insect densities and meteorological factors at some defined locations. These values can be counted to either as weather data or biological cellular automaton to compose state variables.

4.2.3 Data model for insect invasion application

After configuring the above information in section 4.2.1 and 4.2.2 by using the data description schema in figure 3.15, D.1, D.2, D.3, a data model for the insect invasion application is generated as the schema shown in figure 3.16.

4.3 Rice insect pest implementation

4.3.1 Retrieval of environmental data from external sources

Environmental factors are populated from an external source in GRIB format. GRIB is an efficient vehicle for transmitting large volumes of gridded data to automated centers over high-speed telecommunication lines using modern protocols [129]. GRIB is a self-describing, compact structure and is multi-platform compatible.

It is necessary to interpolate parameters from GRIB data to adapt with the cellular structure. The reason is that both the cellular system and GRIB data represent in grid formats but they may have different resolutions. For example, parameters in figure 3.4 are described in (0.5, 0.5) (latitude, longitude) resolution while the cell size is variant depending on the number of pixels chosen for the modeling. Figure 3.4 illustrates the temperature and wind displaying in zyGRIBViewer while figure 4.4 is our spatial cellular system (around 100km²/cell) which presents wind vector in each cell.

Bilinear interpolation [142] (figure 3.19) is used to interpolate parameters on a rectilinear 2D grid. Algorithm in section 3.5.3 is implemented in CUDA to realize this interpolation (Appendix B.1).

CUDA implementation for bilinear interpolation in this phase is necessary since it brings better performance in comparison with PostgreSQL version (Appendix B.2). Table 4.1 shows performances of PostgreSQL and CUDA implementation when taking GRIB data (0.5,0.5) (latitude, longitude) resolution (49 cells) as input in order to output a cellular system consisting of 31311 cells.

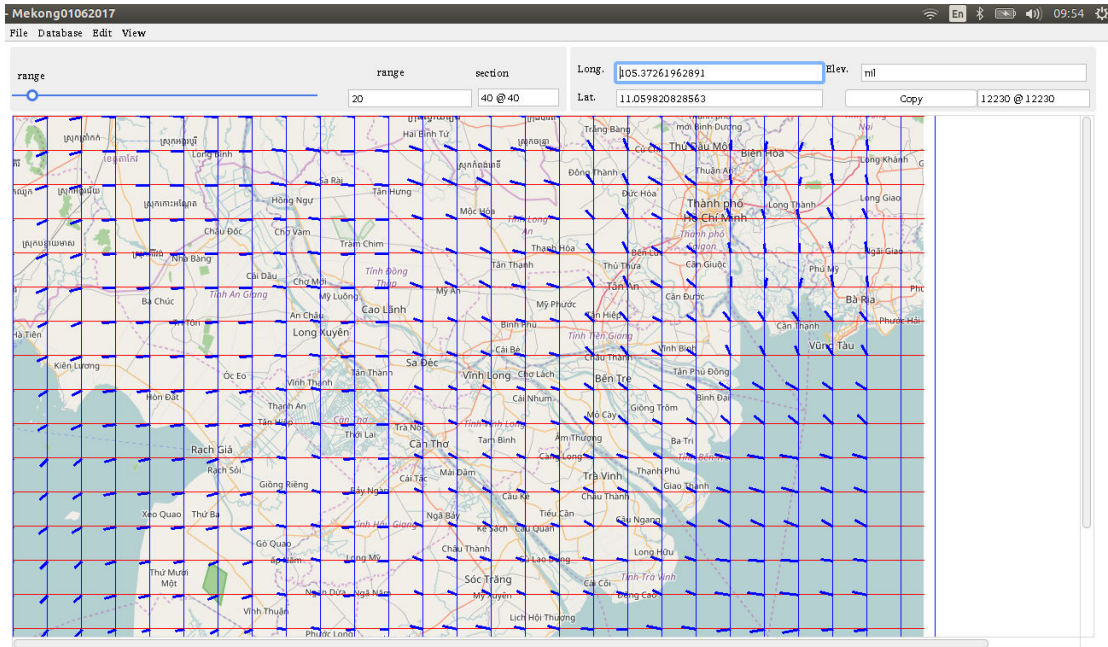


Figure 4.4: Cellular system with (40 pixels, 40 pixels) grid, approximately 100km² per cell, blue arrows representing the wind direction.

Table 4.1: PostgreSQL and CUDA implementation performance in cells definition phase.

| | PostgreSQL | CUDA |
|-------------------------|------------|------|
| Performance (second) | 912 | 27 |

4.3.2 Insect behaviours implementation

Processing model

Each process in the CA machine is presented as a single processing element on GPGPU with bidirectional channels to the other cells. *State* and *Communication links* are two main components for mapping the machine to General-purpose computing on graphics processing units (GPGPU) memory.

1. State. A formal record representing as CUDA structures inside GPGPU memory. Those records are private for each process so they are not supposed to be reached directly by others.
2. Communication links. Links between a process and its neighbors (4 or 8 neighbors). If wind is taken into account, a process has only one link to a leeward neighbor (figure 4.5).

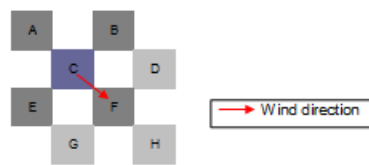


Figure 4.5: Neighbors of a process under wind condition.

To communicate with neighbors, the synchronous model of processing on GPGPU can be resumed as two kernel functions:

1. Buffer writing. Processes write data to output buffers in the shared memory (*Send to buffers*).
2. Buffer reading. Processes read data from their neighbors' output buffers. Next, those buffers are reset for the next execution (*Update states*).

Both reproduction and propagation models (in section 1.2.1) are depicted in these two algorithms.

Send to buffers That task allows transmission of a part of process's values to its buffers. Depending on surrounding conditions, a number of insects is advected to its neighbors. Algorithm is as followed:

```

Input: currentStates_d: Current states of all nodes
Input: buffNodes_d: Buffers of all nodes
Input: channels: Cellular network
Output: A number of insects in a node transmitted to the node's buffers
1  idx ← threadIdx.x + blockIdx.x * blockDim.x
2  if idx < numberOfNodes then
3      curState ← currentStates_d[idx]; curBuff ← buffNodes_d[idx];
        // - Get environment of current day
        // - Calculate current BPH based on preproduction model
        // - Add BPHs number invasion from neighbors
        // - Calculate light trap (if applicable)
4      seed ← curState.seed;
        // Reproduction model
5      pEggs2Nymphs ← _frand(seed, 1.0f, 30.0f) // %eggs 30%
6      pNymphs2Adults ← _frand(seed, 1.0f, 40.0f) // %nymphs 40%
7      numberAdults2Eggs ← _frand(seed, 150.0f, 300.0f) // an adult may give 300 eggs a day
8      pInsects ← _frand(seed, 1.0f, 10.0f) // may 10% insects die per day
9      incAges(curState, curBuffer);
10     layEggs(curState, 4, curBuffer, numberAdults2Eggs);
11     toNymphs(curState, 2, curBuffer, pEggs2Nymphs);
12     toAdults(curState, 3, curBuffer, pNymphs2Adults);
13     foreach child i ∈ channels[idx].read do
14         nodeRead ← i.node
15         identity ← i.identity
16         if nodeRead! = -1 then
17             j ← order_of(i) ∈ channels[idx].read
18             windDirection ← sourceCell.meteo.windDirection
19             if windDirection is on nodeRead then
20                 pMigrations ← _frand(seed, 1.0f, 40.0f) // may 20% insects migrates
21                 moveOut(curState, j, curBuffer, pMigrations);
                    // move pMigration% from idx to curBuffer[j]
22     selfdestroyInsects(curState, curBuffer, pInsects);
23     currentStates_d[idx] ← curState;

```

Algorithm 3: Data transmission to buffers in each node.

Update states This task has responsibility to move data from output buffers to corresponding processes and update their values to compose new states. Details of this

algorithm can be found as followed:

```

Input: currentStates_d: Current states of all nodes
Input: buffNodes_d: Buffers of all nodes
Input: channels: cellular network
Output: Change values of currentState_d
1   $idx \leftarrow threadIdx.x + blockDim.x * blockDim.x$ 
2  if  $idx < numberOfNodes$  then
3       $curState \leftarrow currentStates\_d[idx]; curBuff \leftarrow buffNodes\_d[idx];$ 
4      foreach  $child\ i \in channels[idx].read$  do
5           $nodeRead \leftarrow i.node$  if  $nodeRead \neq -1$  then
6               $j \leftarrow order\_of(idx) \in channels[nodeRead].read$   $buff \leftarrow buffNodes\_d[nodeRead][j]$  // Get buffer j of nodeRead
7               $readBuffer(curState, buff);$ 
              // increase number of insects in curState
8           $writeBuffer(curState, curBuff);$ 
          // deduct number of insects in curState
9       $currentStates\_d[idx] \leftarrow curState;$ 

```

Algorithm 4: Updating node states based on their buffers.

Other operations relating to these two algorithms can be found in appendix E.1.

4.3.3 Simulation result analyse

Some questions can be raised in the simulation result in figure 4.6. How many areas are infected by BPHs? How many percentages of infected areas over the total of rice growing areas? How many of them are hopper-burn? How many of them are tending to become hopper-burn? Technically, these kinds of questions can be solved by combining adjacent cells which have the same infected level into a polygon and maintaining the polygon to the database.

Connected Component Labeling [167][158] (CCL) is a solution for the above combination. It can be done by considering cell system as a map of infected levels from 0 to 5 (0: no infection, 5 hopper burn, table 2.4) and trying to merge the same level connected cells into a polygon (figure 4.7). This algorithm is called right after the execution of the updating cell function in page 95.

Union Find algorithm

The CCL problem can be solved by finding disjoint sets of objects in a graph representing processes in the CA machine. The number of disjoint sets is the number of connected vertices (or connected components) in the graph. For example, figure 4.8 illustrates 10 nodes with 3 connected components which each has 1, 2, 7 nodes, respectively.

Union Find algorithm [175][61] is an algorithm defining two operators in a disjoint graph: Find and Union. Find operator is used to determine whether two objects are in

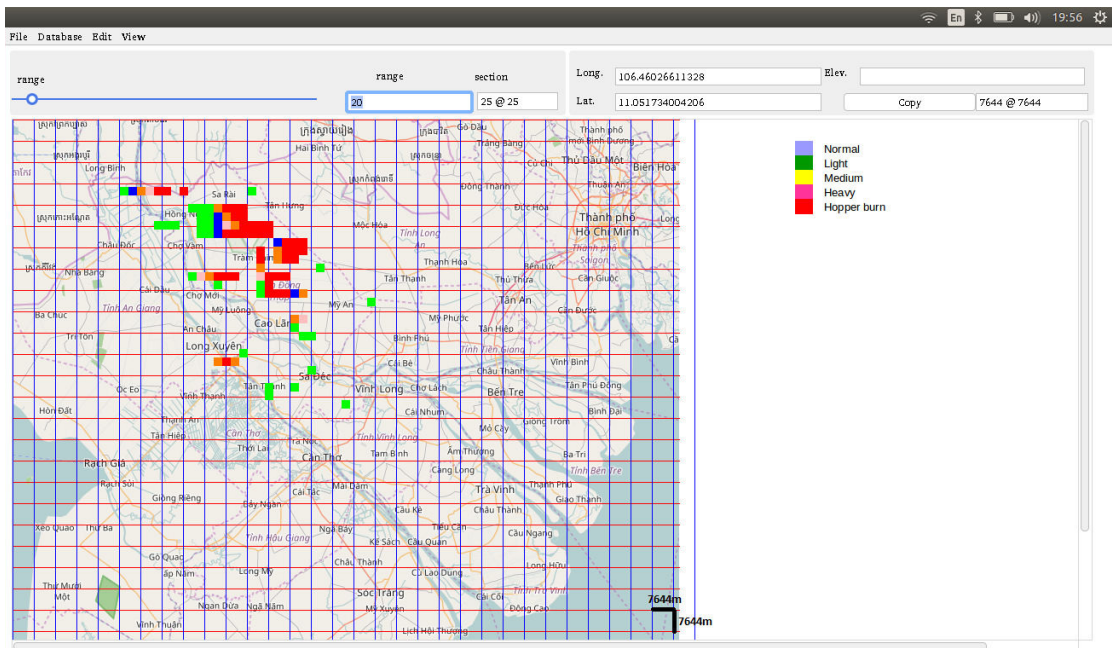


Figure 4.6: Meshed map showing result of the BPH invasion simulation in one step process (grid size = 7644m x 7644m).

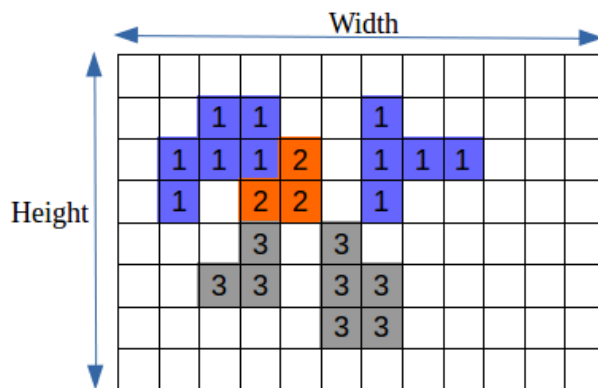


Figure 4.7: Simulation result on a grid represents five distinct polygons a.k.a. area, two of light infection (in blue) 6 and 5 cells, one area of medium infection (in red) and two heavy infection polygons of 3 and 5 cells (in red).

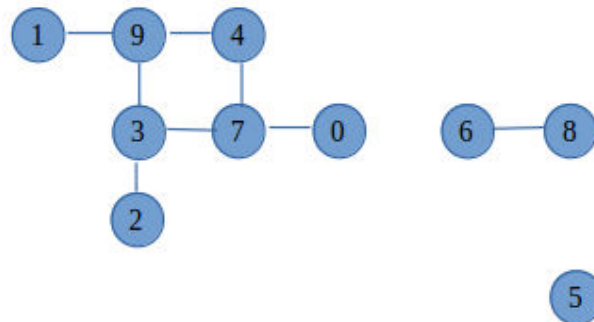


Figure 4.8: CCL problem represented as topology issue, used to find number of connected vertices.

the same sub set while Union operator allows merging two sub sets into one. Thus, CCL problem is solved since those operators have been applied on all nodes of the graph. In the visiting process, nodes belonging to the same set lead to the same root. Therefore, the Find operator checks that two nodes p , q have the same root while Union assigns the root of p to the root of q .

Figure 4.9 depicts an example to solve the CCL in figure 3.19. For any pair p and q , for example, 3 and 9, the root of p is connected to the root of q . After 7 merges, all nodes (10 nodes) in the figure are visited to compose 3 connected vertices. It can be deduced that the number of connected components is $10 - 7 = 3$.

Combine cells into polygons

After executing CCL algorithm, all adjacent cells which share same infection level are combined together to merge into polygons. This mechanism can be done by creating a polygon for each cell and then combining them together by using function `ST_Union` function in PostgreSQL. Algorithm 10 in appendix E.2 is able to do these tasks.

4.4 Experiment

4.4.1 Configuration of experimentations

Simulation of insect pest monitoring application uses data collections in Can Tho city, a typical rice producing city in the Mekong Delta. Can Tho area is divided in 31311

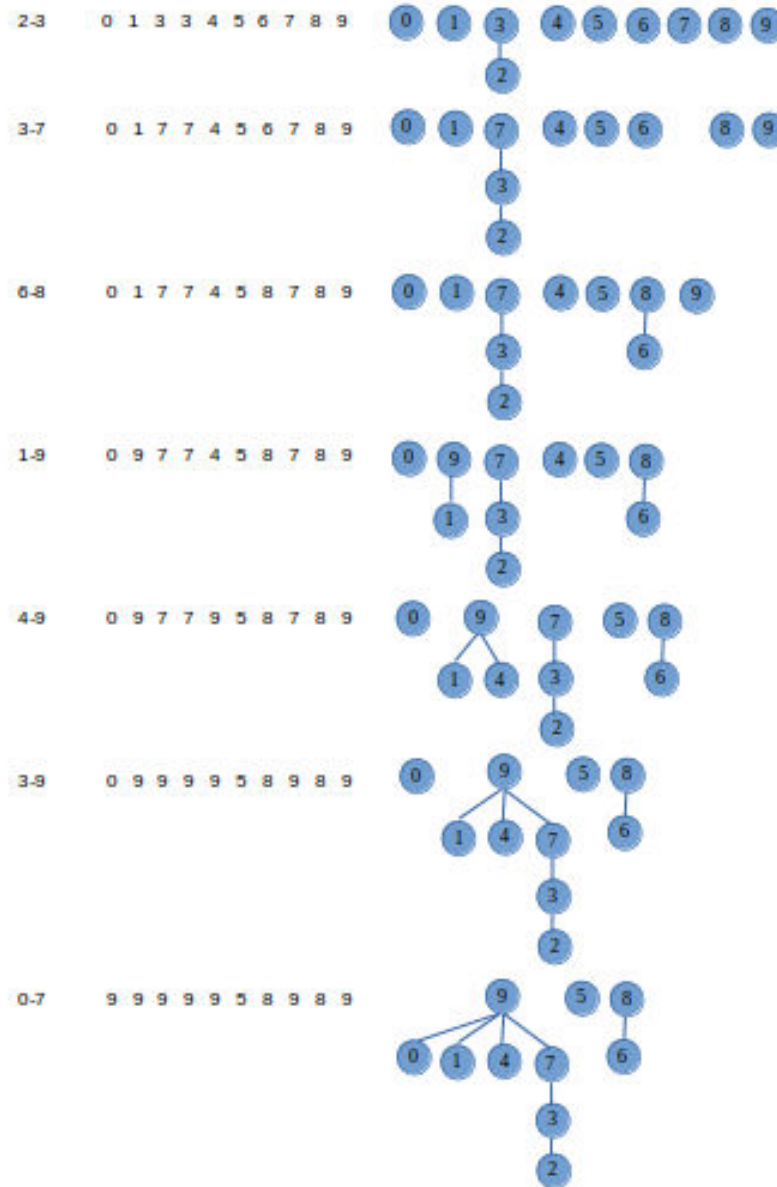


Figure 4.9: Example of Union Find algorithm used to solve CCL of figure 3.19.

cells (Moore neighbors) with approximately 0.18km^2 of surface area for each. The administrative map of that city is used in these experiments where the scale is 1:50000.

Automatic light traps (circles in figure 4.10) provides a data collection including insect densities. Environmental factors are populated from GRIB data.

To visualize infected BPHs in the insect monitoring application, a color table is used to illustrate a warning level of BPHs based on hoppers density in rice fields [137]. This correspondence is shown in table 4.2.

Table 4.2: Ascending levels of infested BPHs in rice fields.

| BPH density (BPHs/m²) | Color | Meaning |
|---|------------------|------------------|
| <500 | rgb[135,206,250] | Normal infection |
| 500 -<2500 | rgb[0,255,0] | Light infection |
| 2500 -<5000 | rgb[255,255,0] | Medium infection |
| 5000 -<10000 | rgb[255,105,180] | Heavy infection |
| ≥ 10000 | rgb[255,0,0] | Hopper burn |

4.4.2 Scenario 1: Observing hoppers at a defined location

This scenario allows observation of the reproduction phase of hoppers at a defined location. In this scenario, some communes in Can Tho lightly suffer from hoppers (light infection color of rice fields and warning level of sensor stations in figure 4.10) while the rest are not cultivated yet.

Figure 4.11 describes the hoppers infection in Can Tho city at the day 3, 4, 6, 7. At the day 3, most of the experimental communes suffer from heavy infection of BPHs. Hoppers density reaches the peak point at the day 4 and starts decreasing a few days later. During the day 7, normal infections appear in some communes although other are still light, medium or even heavy (warning level of sensor stations in figure 4.12). However until the day 9, hoppers seem not to appear in Can Tho city.

This scenario depicts the reproduction of hoppers in Can Tho city. Initially, experimental communes are infested lightly due to hoppers. By the time passing, hoppers are growing and become adults. At the adult phase, hoppers can propagate to other places due to wind intensity, however, these new locations do not have rice, therefore,

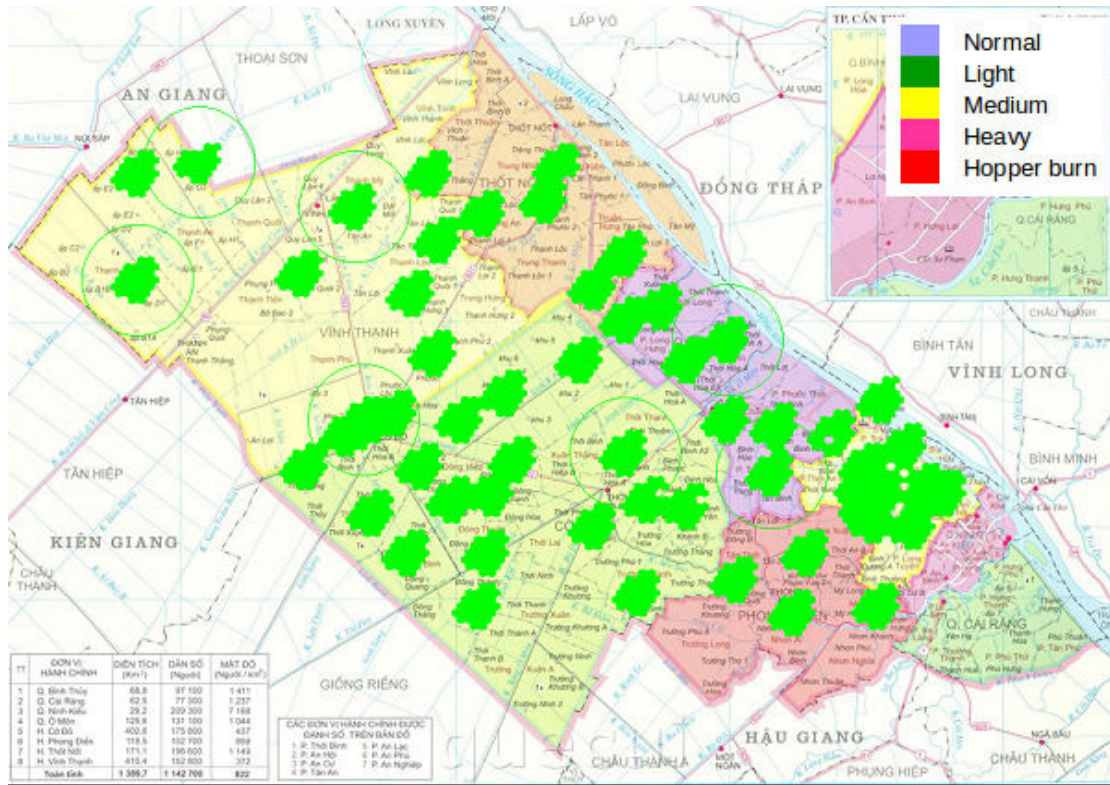


Figure 4.10: Light infections at some communes in Can Tho city.

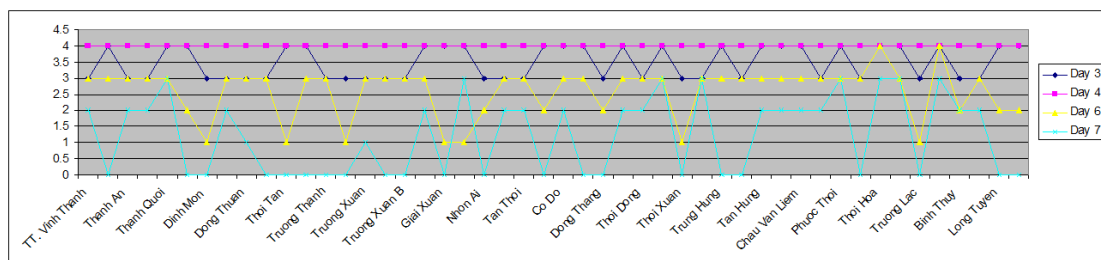


Figure 4.11: Hoppers infection level in Can Tho city for days 3, 4, 6, 7 at several locations.

propagated hoppers die because of food lacking. Densities of hoppers at experimental communes decline gradually and return at normal infestation level around 6 to 7 days after the peak day of hoppers.

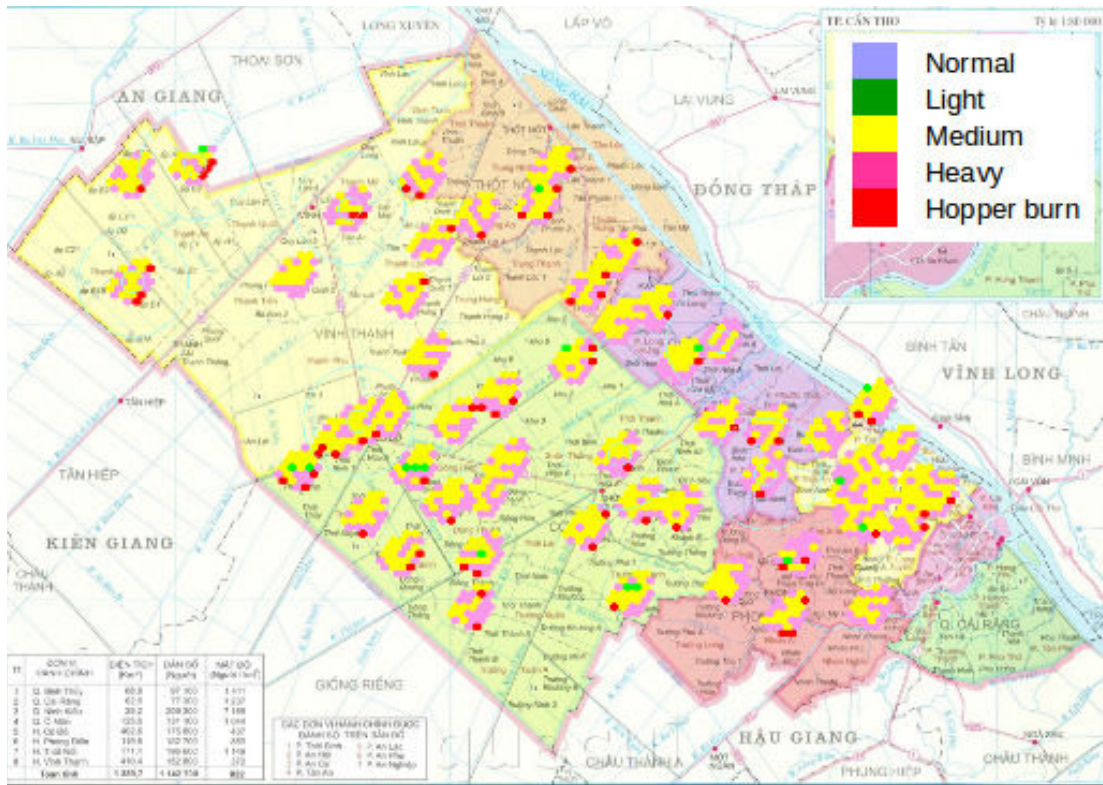


Figure 4.12: Map of Hoppers infection in Can Tho city at day 7.

The hopper reproduction in the scenario shows how 'Escape Strategy' (or 'Chien Luoc Ne Ray' in Vietnamese) does to confront with hoppers [71]. This strategy is done by monitoring historical light traps data through several years to recognize the trend of hoppers migrations. Next, crops are sown after the peak season of BPHs. When the next generation of hoppers comes (28 days later), the rice is strong enough to resist to hoppers. In this case light trap stations can help to sense the surrounding conditions and transmit data to the database center (schemas as in figure 3.16) for storage so the peak of hoppers can be identified later.

4.4.3 Scenario 2: Hoppers propagation due to wind

This scenario assumes that the Thoi Lai district is an infection source with lightly infestation in almost its communes (figure 4.13). A current unique sensor at Dinh Mon,

Thoi Lai provides following meteorological data: wind velocity: 5km/h, wind direction: North North Wind - NNW (from Dinh Mon toward Phong Dien) and also indicates the light infection at the rice field in this commune.

| Day/Commune | THOI LAI DISTRICT | | | | | | | | | | | | | PHONG DIEN DISTRICT | | | |
|-------------|-------------------|----------|-----------|------------|-----------|----------|------------|--------------|--------------|-------------|---------------|---------------|------------|---------------------|----------|---------|------------|
| | Thoi Lai | Dinh Mon | Dong Binh | Dong Thuan | Tan Thanh | Thoi Tan | Thoi Thanh | Truong Thanh | Truong Thang | Truong Xuan | Truong Xuan A | Truong Xuan B | Xuan Thang | Truong Long | Tan Thoi | Nhon Ai | Nhon Nghia |
| 0 | L | L | L | L | L | L | L | L | L | L | L | L | L | * | * | * | * |
| 2 | B | B | B | H | B | B | B | H | H | H | H | H | B | N | N | * | * |
| 4 | B | B | B | B | B | B | B | B | B | B | B | B | B | L | N | N | N |
| 7 | B | N | N | N | N | B | N | B | N | N | B | B | N | H | L | N | N |

*: No Infected N: Normal L: Light M: Medium H: Heavy B: Burn

Figure 4.13: Hoppers infestation in Thoi Lai and Phong Dien in 7 days.

Due to wind, hoppers can propagate to leeward fields. At the day 3, the whole Thoi Lai is burned and a part of Phong Dien is lightly infected (figure 4.13, 4.14). At the day 7, the light trap at Dinh Mon still gives burn warning although other communes in this district become normal. However, the area of infestation in Phong Dien district is broaden from Truong Long, Tan Thoi, to Nhon Ai, Nhan Nghia. BPHs spread over these communes from heavy infection in Truong Long to light ones (almost normal) in Nhon Ai, Nhon Nghia (figure 4.15). Indeed, under the wind, the commune Truong Long, Phong Dien is the leeward of the commune Truong Thanh, Thoi Lai. Similarly, the commune Tan Thoi, Phong Dien is the leeward of the commune Dinh Mon, Thoi Lai. Same situations can be found in Nhon Ai, Tan Thoi and Nhon Nghia.

The light trap in Dinh Mon uses a led 4W, 220V/50Hz as the light bulb. Although we never measure the sensing range in practice (formular 2.1), assumptions are made to calculate the coverage area of the trap.

Assume that the light bulb has sensing radius of 830m (value in FullMoon in [43]), in ideal case, the sensing field has an area: $S = \pi r^2 = 2163146m^2 \approx 2.16km^2$.

Thoi Lai district (Dinh Mon is a town of it) has 20,345.16 ha rice field ¹. Therefore, the minimum number of equipments to monitor the whole area is $N \geq 94$ traps.

Nevertheless, in practice, distribution of dense traps is not necessary since one device can be used as an indication value for insect situation of a wide area. By observing historical trap values, Escape Strategy is applied for planning crop calendar which can avoid insect peak days.

The deployment of Rothamsted light trap network in Britain is not dense as well. Currently, the network has 87 sites in operation [202], where the country has $\approx 242,000km^2$

¹<http://cantho.gov.vn/wps/portal/thoilai>

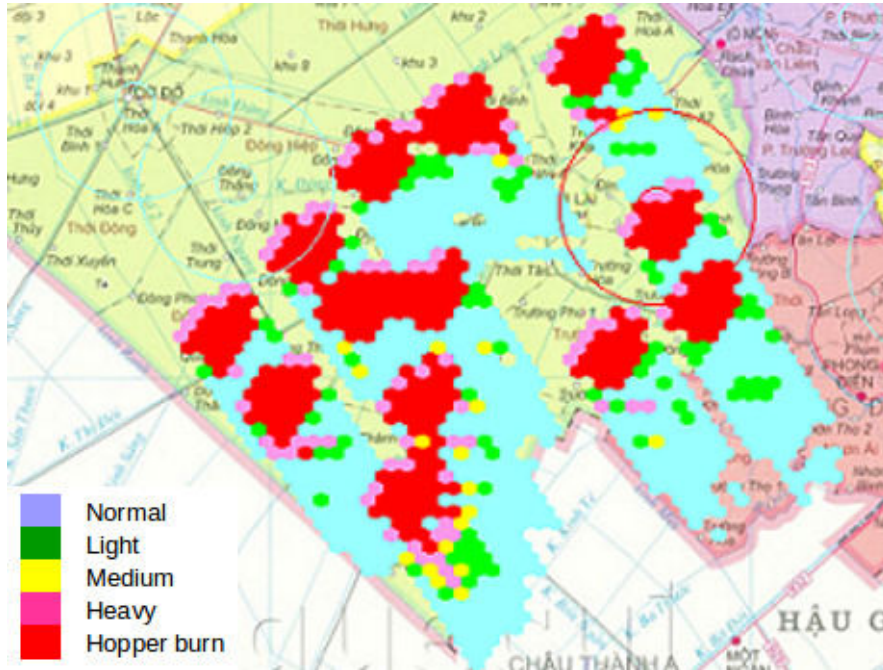


Figure 4.14: Hoppers infection in Thoi Lai and Phong Dien at day 3.



Figure 4.15: Map of leeward communes in Phong Dien district, arrow represents the average wind direction

of surface area ².

4.5 Summary

The chapter presents the model of a rice insect pest application based on data oriented using cellular automata. Environmental, biological factors and spatial cellular system are integrated into a unit to form the static part while the dynamic aspect, transition rule, is applied locally to compose a CA machine for insect monitoring.

The simulation shows cyber physical relations in the insect monitoring application. Feedback loops are found with the collaboration between physical phenomena (insect, rice) and trap devices. Current situation of insects is measured by sites and vice versa, trap stations attract and kill insects to reduce their populations. It reflects the local cyber physical aspect. On the other hands, 'Escape Strategy' is a kind of solution to face with pest and it illustrates the global interaction of an environment machine.

²<https://www.mapsofworld.com/united-kingdom/britain/>

5

Conclusions

5.1 Thesis results

As described in section 1.2.3, goal of this thesis is about an early warning system of which data collection is from sensor stations. The system consists of following components which are done during the thesis time:

- Sensor stations. They measure environmental factors and classify/count insect densities automatically. The principle is done based on vision sensors of which principles are depicted in chapter 2. These sites constitute an automatic light trap network in Mekong Delta.
- Modeling and simulation. Simulation is done with the assistance of environment machine (chapter 3) which is built on cellular automata modeled synchronous system. The machine consists of processes which are able to execute the same program concurrently.

- Information system for environment machine. The machine requires collecting and analyzing of data from different sources. As a result, the architecture proposed in figure 3.11 and the schema presented in figure 3.15 are able to adapt with data requirements in environmental monitoring applications.

Those components emerge local/global cyber physical system concepts and are supported by developed tools.

5.2 Local vs. Global Cyber Physical System

A Cyber Physical System (CPS) [112] is a system of collaborating computation and physical processes. Embedded computers and sensor networks monitor the physical processes, essentially with feedback loops in respect of interaction between processes and computations. In physical world, time passage and concurrency are two core characteristics.

An environmental monitoring system fits into a CPS framework. Environmental factors become physical entities while the observation network is the computation. For example, sensor nodes, are able to monitor the surrounding conditions and with the assistance of data collection over time, people can make decision relating to their situations. These elements build together a temporal loop between physical entities and numerical computation.

That study goes through the Cyber Physical System concept from local to global scale. Figure 2.1 and 3.1 describe those scales.

Observation systems are distributed sensing devices which collect information locally, elaborate diagnostics on physical process, and report those diagnostics. Each of them permits two-ways interactions between physical world and sensing devices. Automatic trap station, a practical application of that principle to monitor insects in rice fields, is studied to show that two-ways interaction. Fisheye lens camera in the trap periodically classifies/measures densities of insects while light bulb and other components are capable of changing insects behaviors by attracting them to the light. Ten automatic trap stations currently in test in Mekong Delta are the final product of that study.

Connected observation systems compose a global Cyber Physical System which shows feedback loops between physical and computation world. Natural phenomena, representation of physical world, occur and are measured by sampling sensing devices. The information system behind the CPS is able to control and reduce impact of phenomena.

Combination of automatic trap stations is a representation of that relationship. Insects breed quickly and spread in order to find food to eat but acting like that action they destroy rice fields. Their quantities are periodically estimated by trap stations running. And observation of long series of data allow to make the appropriate choice to face insect pest. 'Escape Strategy' is one of those decisions based on data analysis.

5.3 NetGen and Pickcell

Tools for Cyber Physical System (especially those presented in chapter 3 & chapter 4) are implemented as update versions of NetGen and Pickcell [15]. NetGen is a framework developed in the group Wireless Sensor Network, University of Western Brittany, France by Professor Bernard Pottier with initial purposes design and validation of networks under a practical environment and using graphical interface. Pickcell is a module of NetGen to create spatial cellular systems from GIS data. In general, the framework provides these common characteristics:

1. Practical system description based on geometry. As example, from a map, one can decide sensor locations by taking into account physical considerations, decide on a wireless technology, and infer workable communication links.

Description can be achieved based on maps, or pictures. Alternatively, generators allow to produce random distributions of different characteristics. An input text format allows exchange of network topologies with external tools.

2. Behaviour description. As example, nodes will execute programs to control locally a physical phenomenon and to contribute to activities of the distributed system, such as collecting, transforming data or sending alerts.
3. The core of NetGen is Pickcell tool which allows modeling spatial spaces as cellular automata. Figure 3.8 shows an interface of PickCell where the spatial space is divided in a 25*25 resolution cells.

Cyber Physical System characteristics can be found in NetGen framework. From the CPS concept map (figure 5.1) provided by Edward A. Lee, University of California, Berkeley, main features can be recognized:

¹Source: <http://cyberphysicalsystems.org/>

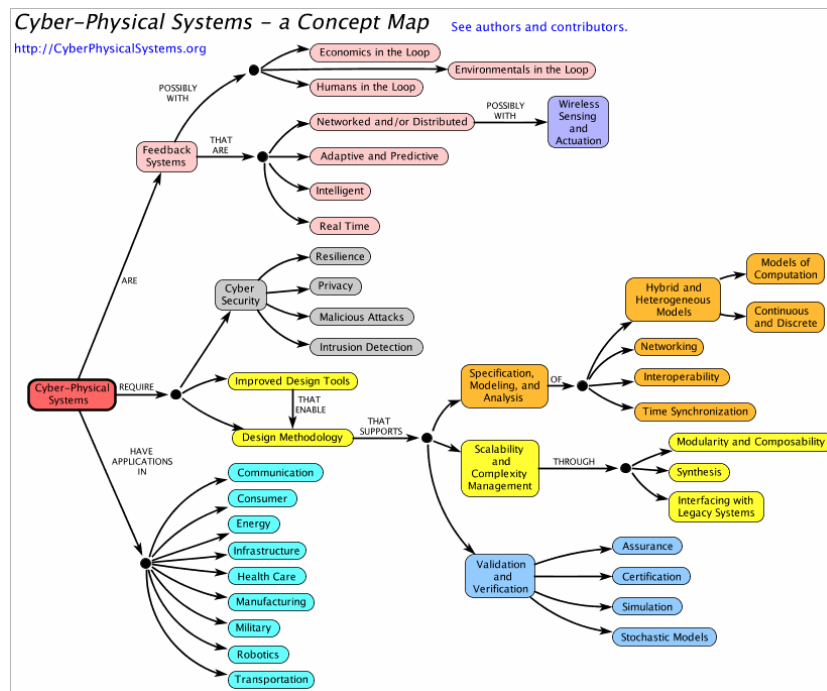


Figure 5.1: Concept map of Cyber Physical System¹.

- Application fields. The framework covers a lot of situations, from the nano scale to the universe scale and lots of domains such as distributed sensing, distributed computing, communication systems, environment modeling, bio-systems. For example, the context addressed in this manuscript is environmental monitoring as well as distributed sensing devices to measure and control the environment.
- Networks and Distributed computations. Important focus of the framework is on wireless sensor networks design in regards to practical situations in environmental field.
- Feedback loops. Organization of data collection in the framework allows understanding and control the behavior of environment. The case study shows that characteristic.
- Specification, modeling, and analysis. Pickcell tool in the framework is able to model physical world as a cellular automata synchronous system that ensures interoperability and communication between processes. Those elements are main problems of Cyber Physical Systems.

5.4 Future plans

Several plans are taken into considerations to continue the development of sensing machines particularly for insect pest mitigation. Three of them are presenting below including sensors type, validation methods and computation capacity:

- Camera with parallel processor. Despite the fact that fisheye lens camera with embedded processor is able to classify and count insect densities in a trap site, it is unable to do the same in more complicated cases such as tracking, moving objects recognition. Camera *integrating data acquisition and processing unit* is the best choice for those more complex usages.
- Validation and verification. Validation is the process of checking whether a design captures the customer's needs whereas verification determines that a design meets requirements, specifications. If requirements, specifications are given as formal languages, the concurrent model using CA machine may be possible to be automated the verification (also know as formal verification). Meanwhile, CA models generated by Pickcell software will be included in a validation process of which verification is a part. That process can not be automated, but possibly by simulation.
- Cellular automata on super computers equipped with large amount of nodes. CA synchronous model is adjusted so that it can be implemented on many computers to process big data. Message Passing Interface (MPI) [57] is capable to generate solution for that.



Appendix A

A.1 Validation methods

A.1.1 Normal distribution

Let x depicts sizes of image objects which have the same distance to the image center. Because these objects have the same real size, their images represent random variables which converge in distribution to the normal, or normal distribution (Gaussian distribution) [50].

The probability density function (figure A.1) of a normal distribution is calculated as:

$$f(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where:

- μ : mean or expectation of the distribution
- σ : standard deviation
- σ^2 : variance

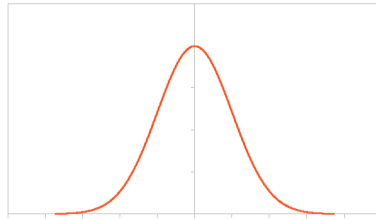


Figure A.1: Probability density function of a normal distribution.

A.1.2 Shapiro-Wilk test

The Shapiro-Wilk [161] test is a normality test to detect if a sample x_1, \dots, x_n coming from a normal distribution. The test statistic:

$$W = \frac{(\sum_{i=1}^n a_i x(i))^2}{\sum_{i=1}^n x_i - \bar{x}}$$

Where

- $x(i)$: the i^{th} order statistic, or the i^{th} smallest number in the sample.
- \bar{x} : the sample mean
- a_i calculated by:

$$(a_1, \dots, a_n) = \frac{m^T V^{-1}}{\sqrt{m^T V^{-1} V^{-1} m}}$$

where:

- $m = (m_1, \dots, m_n)^T$
- m_1, \dots, m_n : expected values of the order statistics independent and identically distributed random variables sampled from the standard normal distribution.
- V : the covariance matrix of those order statistics.

Hypothesis of normality is rejected if the p-value is less than or equal 0.05. A failed test states that at 95% confidence data does not fit to the normal distribution while passing the test just states that no significant departure from normality was found.

A.1.3 Kolmogorov-Smirnov test

Kolmogorov-Smirnov test [27] [135] tests a sample from a population with a specific distribution. It is based on the empirical distribution function (ECDF). Given N ordered data points Y_1, Y_2, \dots, Y_N , the ECDF is defined as:

$$E_N = \frac{n(i)}{N}$$

where $n(i)$ is the number of points less than Y_i and the Y_i are ordered from smallest to largest value.

The test is defined by:

- H_0 : the data follows a specified distribution (e.g normal).
- H_a : the data does not follow the specified distribution.
- Test Statistic:

$$D = \max_{1 \leq i \leq N} \left(F(Y_i) - \frac{i-1}{N}, \frac{i}{N} - F(Y_i) \right)$$

- P value (significance level): to see how much different data collected in one situation (is compared to data collected in a different situation. If p value is smaller or equal 0.05, the hypothesis of distribution (normality) is rejected.

A.2 Validation results

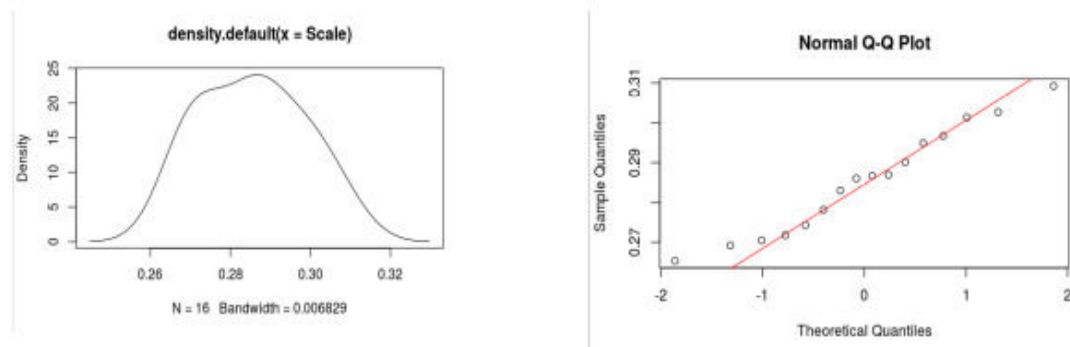


Figure A.2: Some statistical results of objects in group 1: Probability density function, quantile-quantile (q-q) plot of object scales in the image.

The Shapiro-Wilk normality test which is applied on these measurements gives the test statistic $W = 0.96391$ and the significance level $p_value = 0.7329$.

Table A.1: Expectation and deviation of measured sizes (pixels and scale) of objects in group 1 (distance around 73.32 pixels from the image center).

| Object | Distance from the center | Number of pixels | Scale pixels/mm² |
|---------------|---------------------------------|-------------------------|------------------------------------|
| 28 | 69.9978 | 175.733 | 0.286034 |
| 34 | 70.0073 | 173.253 | 0.290128 |
| 24 | 70.6433 | 169.415 | 0.296701 |
| 38 | 70.7466 | 175.167 | 0.286957 |
| 42 | 71.5176 | 175.337 | 0.286679 |
| 21 | 71.6129 | 162.553 | 0.309224 |
| 17 | 72.1238 | 170.446 | 0.294905 |
| 45 | 72.8646 | 180.712 | 0.278152 |
| 43 | 73.3217 | 186.706 | 0.269223 |
| 14 | 74.3196 | 166.072 | 0.302673 |
| 40 | 75.1910 | 189.445 | 0.265330 |
| 35 | 75.7579 | 183.239 | 0.274317 |
| 16 | 76.0527 | 166.813 | 0.301328 |
| 25 | 76.0905 | 184.948 | 0.271782 |
| 20 | 76.2874 | 177.625 | 0.282986 |
| 30 | 76.5369 | 185.829 | 0.270494 |

B

Appendix B

B.1 Bilinear interpolation implementation in PostgreSQL

```
Input: Geographic position (latitude, longitude) of a cell  
Output: uWind, vWind, temperature, humidity, precipitation, pressure at this position  
  
Drop Table If Exists TestWeather;  
--TestWeather(x int, y int, latitude numeric, longitude numeric,  
--  uValue numeric, vValue numeric,  temperature numeric,  
---  humidity numeric, precipitation numeric, pressure numeric)  
Create Table TestWeather As Select * From getWeatherData();  
Select into minLat, minLon Min(Latitude)*10, Min(Longitude)*10  
From TestWeather;  
m_iLat := lat*10; m_iLon := lon*10;  
xPos := (m_iLat-minLat)/5; yPos := (m_iLon-minLon)/5;  
---- x increases from bottom to top, y from left to right ----  
  
-- point 1  
Select into u1, v1, t1, h1, pp1, p1 uValue, vValue, temperature,  
humidity, precipitation, pressure  
from TestWeather  
Where x = (cast(xPos as int)) AND y = cast(yPos as int);
```



```

-- point 2
Select into u2, v2, t2, h2, pp2, p2 uValue, vValue, temperature,
humidity, precipitation, pressure
from TestWeather
Where x = (cast(xPos as int) + 1) AND y = cast(yPos as int);
-- point 3
Select into u3, v3, t3, h3, pp3, p3 uValue, vValue, temperature,
humidity, precipitation, pressure
from TestWeather
Where x = (cast(xPos as int)+1) AND y = (cast(yPos as int)+1);
-- point 4
Select into u4, v4, t4, h4, pp4, p4 uValue, vValue, temperature,
humidity, precipitation, pressure
from TestWeather
Where x = cast(xPos as int) AND y = (cast(yPos as int) + 1);

--- Bilinear interpolation
tt1 := (trunc(yPos) + 1 - yPos)*u2 + (yPos - trunc(yPos))*u3;
tt2 := (trunc(yPos) + 1 - yPos)*u1 + (yPos - trunc(yPos))*u4;
uWind := (trunc(xPos) + 1 - xPos)*tt1 + (xPos - trunc(xPos))*tt2;

tt1 := (trunc(yPos) + 1 - yPos)*v2 + (yPos - trunc(yPos))*v3;
tt2 := (trunc(yPos) + 1 - yPos)*v1 + (yPos - trunc(yPos))*v4;
vWind := (trunc(xPos) + 1 - xPos)*tt1 + (xPos - trunc(xPos))*tt2;

tt1 := (trunc(yPos) + 1 - yPos)*t2 + (yPos - trunc(yPos))*t3;
tt2 := (trunc(yPos) + 1 - yPos)*t1 + (yPos - trunc(yPos))*t4;
temp := (trunc(xPos) + 1 - xPos)*tt1 + (xPos - trunc(xPos))*tt2;

tt1 := (trunc(yPos) + 1 - yPos)*h2 + (yPos - trunc(yPos))*h3;
tt2 := (trunc(yPos) + 1 - yPos)*h1 + (yPos - trunc(yPos))*h4;
hum := (trunc(xPos) + 1 - xPos)*tt1 + (xPos - trunc(xPos))*tt2;

tt1 := (trunc(yPos) + 1 - yPos)*pp2 + (yPos - trunc(yPos))*pp3;
tt2 := (trunc(yPos) + 1 - yPos)*pp1 + (yPos - trunc(yPos))*pp4;
prep := (trunc(xPos) + 1 - xPos)*tt1 + (xPos - trunc(xPos))*tt2;

tt1 := (trunc(yPos) + 1 - yPos)*p2 + (yPos - trunc(yPos))*p3;
tt2 := (trunc(yPos) + 1 - yPos)*p1 + (yPos - trunc(yPos))*p4;
press := (trunc(xPos) + 1 - xPos)*tt1 + (xPos - trunc(xPos))*tt2;

```

Algorithm 5: Bilinear interpolation to calculate environmental parameters at a geographic location.

B.2 Bilinear interpolation implementation in CUDA

Input: A(lat1, lon1), B(lat2, lon2)

Output: Distance between A & B

```
__device__ double getDistance(double lat1, double lon1,
    double lat2, double lon2)
{
    double earthRadiusKm = 6371;

    double dLat = degreesToRadians(lat2-lat1);
    double dLon = degreesToRadians(lon2-lon1);

    lat1 = degreesToRadians(lat1);
    lat2 = degreesToRadians(lat2);

    double a = sin(dLat/2) * sin(dLat/2) + sin(dLon/2) *
        sin(dLon/2) * cos(lat1) * cos(lat2);
    double c = 2 * atan2(sqrt(a), sqrt(1-a));
    return earthRadiusKm * c;
}
```

Algorithm 6: Distance between 2 points in the earth.

Input: Climate data in resolution 0.5*0.5 (latitude, longitude) from GRIB source

Output: New cellular system with smaller resolution

```

__global__ void doInterpolation(ClimateData *data_d, int nSize,
CellArray *cell_d, int nCells)
{
int idx = threadIdx.x + blockIdx.x * blockDim.x;

if (idx < nCells)
{
CellArray cell = cell_d[idx];
double lat = cell.position.latitude*10;
double lon = cell.position.longitude*10;
double minLat,minLon,maxLat,maxLon;
int minX, minY, maxX, maxY, minPos, maxPos, pos;
double u_B3, v_B3, u_B1, v_B1, u, v;
double u1,v1,u2,v2,u3,v3,u4,v4;

bool found;

//Min value
minLat = ((int)lat)/5*5;
minLon = ((int)lon)/5*5;
maxLat = ((int)lat)/5*6;
maxLon = ((int)lon)/5*6;

minPos = getPosition(maxLat,minLon,data_d,nSize, found);
//max value
maxPos = getPosition(minLat,maxLon,data_d,nSize, found);

minX = data_d[minPos].x;
minY = data_d[minPos].y;
maxX = data_d[maxPos].x;
maxY = data_d[maxPos].y;

lat = lat/10;
lon = lon/10;

```

```

// Pos 1 (minLat, minLon) A1 , Pos 2 (minLat, maxLon) A2
// Pos 3 (maxLat, maxLon) A3 , Pos 4 (maxLat, minLon) A4
// Pos (lat, lon) I
// B1 (minLat, lon), B2 (lat, maxLon)
// B3 (maxLat, lon), B4 (lat, minLon)
u4 = data_d[minPos].uWind;
v4 = data_d[minPos].vWind;
u2 = data_d[maxPos].uWind;
v2 = data_d[maxPos].vWind;
//get values of remaining
pos = getPosition(minLat,minLon,data_d,nSize, found);
u1 = data_d[pos].uWind;
v1 = data_d[pos].vWind;
pos = getPosition(maxLat,maxLon,data_d,nSize, found);
u3 = data_d[pos].uWind;
v3 = data_d[pos].vWind;
//Step 1
// a
u_B3 = getDistance(maxLat, maxLon, maxLat, lon)/getDistance(maxLat,maxLon,maxLat,minLon)*u4 +
getDistance(maxLat,lon,maxLat,minLon)/getDistance(maxLat,maxLon,maxLat,minLon)*u3;
v_B3 = getDistance(maxLat, maxLon, maxLat, lon)/getDistance(maxLat,maxLon,maxLat,minLon)*v4 +
getDistance(maxLat,lon,maxLat,minLon)/getDistance(maxLat,maxLon,maxLat,minLon)*v3;
//b
u_B1 = getDistance(minLat,maxLon,minLat,lon)/getDistance(minLat,maxLon,minLat, minLon)*u1 +
getDistance(minLat,lon,minLat,minLon)/getDistance(minLat, maxLon, minLat, minLon)*u2;
v_B1 = getDistance(minLat,maxLon,minLat,lon)/getDistance(minLat,maxLon,minLat, minLon)*v1 +
getDistance(minLat,lon,minLat,minLon)/getDistance(minLat, maxLon, minLat, minLon)*v2;
//Step 2
u = getDistance(minLat, lon,lat,lon)/getDistance(minLat, lon, maxLat, lon)*u_B3 +
getDistance(lat,lon,maxLat, lon)/getDistance(minLat, lon, maxLat, lon)*u_B1;
v = getDistance(minLat, lon,lat,lon)/getDistance(minLat, lon, maxLat, lon)*v_B3 +
getDistance(lat,lon,maxLat, lon)/getDistance(minLat, lon, maxLat, lon)*v_B1;
cell.weatherArray.ugrd = u;
cell.weatherArray.vgrd = v;
cell_d[idx] = cell;
}
}

```

Algorithm 7: Bilinear interpolation implementation in CUDA.

C

Appendix C

C.1 Sensing machine modeling in NetGen

Principles of a global sensing machine can be modeling in NetGen [15], a framework is dedicated to WSN design and validation in regards of practical situations in the environment. Workflow for generating and executing the machine [182][107] is shown in figure C.1.

The design and implementation of the machine contain three important parts: data structure, states and behaviors. Firstly, data structure (cells in figure 3.8) is generated from geographic data using PickCell tool in NetGen [15]. Next, states and behaviors are implemented in parallel programming languages to respect synchronous features in the machine.

For example, cells under a synchronous system structure is generated into CUDA code [24] as following:

```
#define NODE_NUMBER 31311
#define MAX_FANOUT 8

typedef struct s_mapped{
    int node;
    int canal;
    int identity;
} mapped;
typedef struct s_canaux{
```

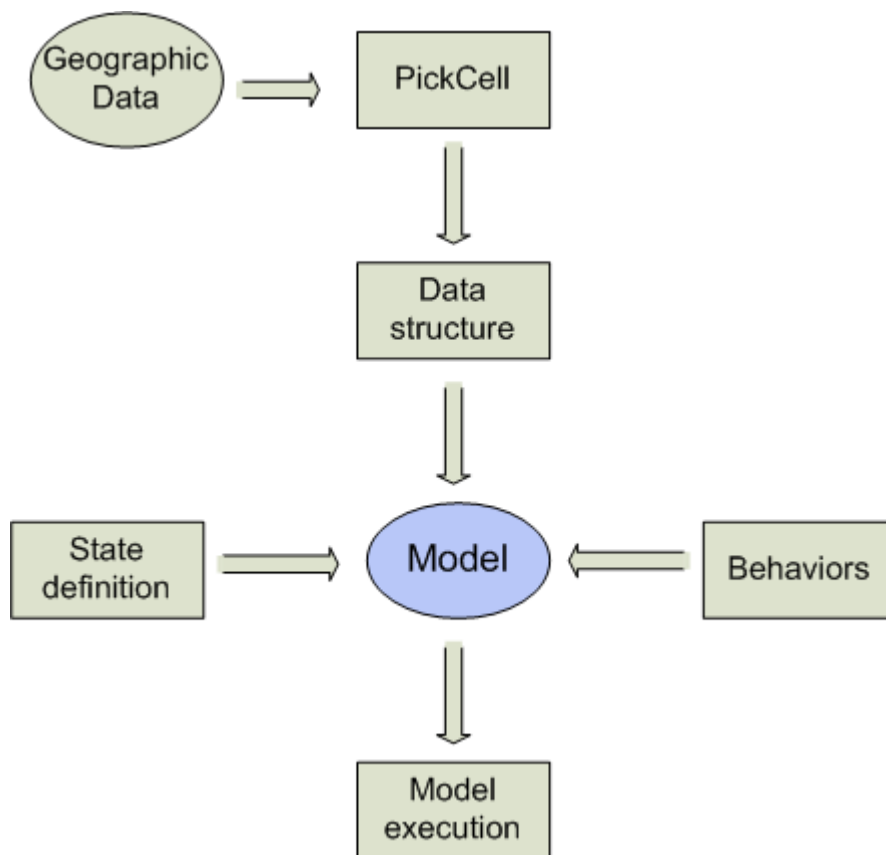


Figure C.1: Work flow for modeling insect physical and WSN system.

```
        int nbOut;
        int nbIn;
        int nbDyn;
        mapped write [MAX_FANOUT];
        mapped read [MAX_FANOUT];
} canaux;
typedef struct {
        int x;
        int y;
        int range;
} node_param;
```

Each node of the network represents a process in the machine. That process maintains a structure to store its states by the time passing. Definition of the structure is variant and depends on which on application the machine is applied.

D

Appendix D

D.1 Data descriptions for meteorological, object and sensor data

Those descriptions are depicted in figure D.1, D.2 and D.3.

D.2 Workflow for developing an environmental monitoring application

The schema in figure 3.15 allows the conception of a workflow (figure D.4) for develop environmental monitoring applications such as insect invasion, flooding, rain falls. Data definition step helps to define meteorological factors, objects as well as sensor nodes which involve the phenomena. These elements of data will be generated to parallel programming languages for behavior implementations. Those implementations, in general sense, are done as illustration in section 4.3. Finally, simulation as well as other data can be queried for user purposes such as aggregation, tendency prediction.

For example, to simulate insect invasion, after defining environmental factors, the sensing machine has a structure similar to the one presented in figure 3.9. Next, the cellular automaton and relating data are generated into two CUDA files: one file for cellular automaton description (listing D.1) and another for relating data (listing D.2).

```
#define NODE_NUMBER 31311
```

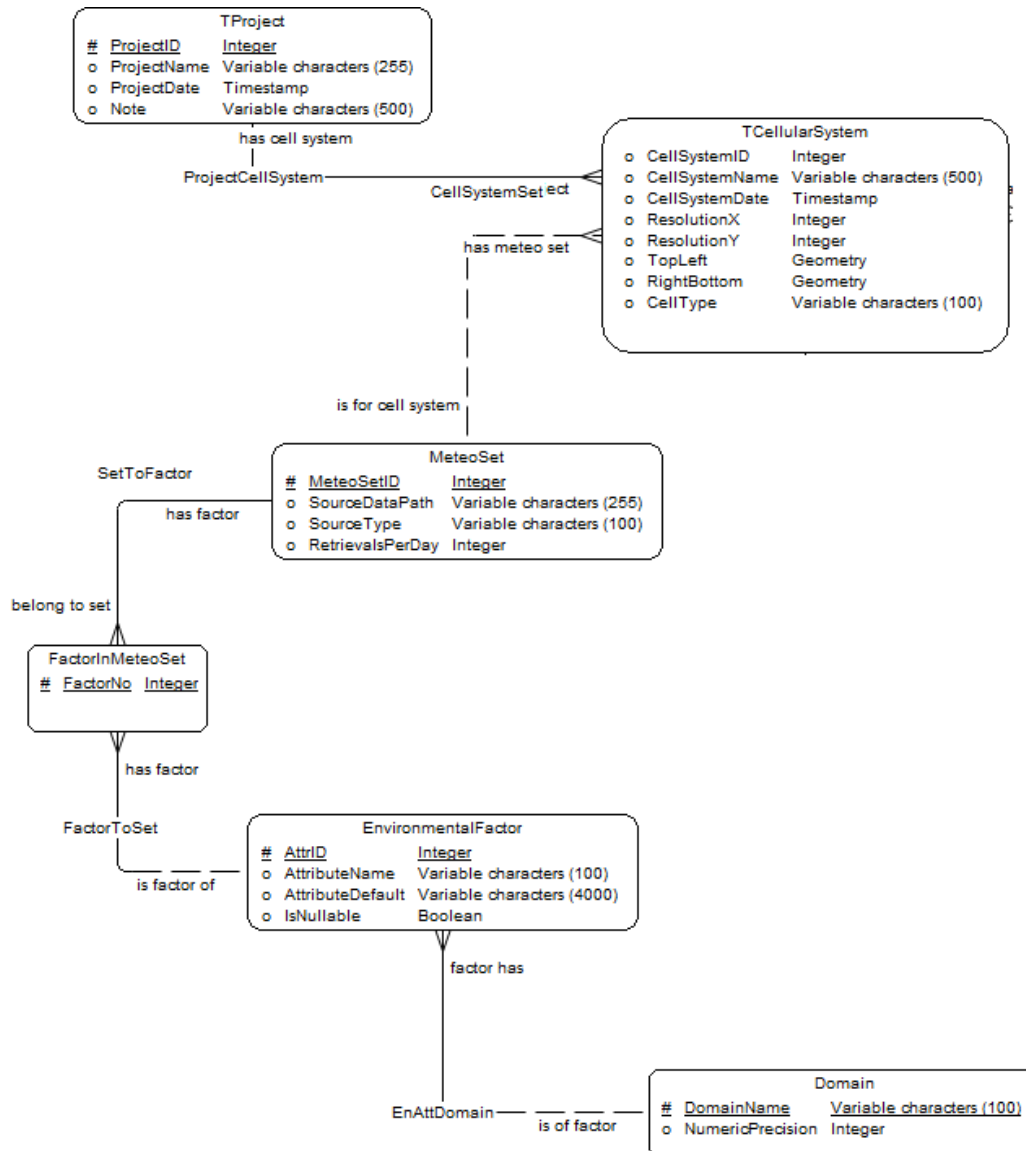
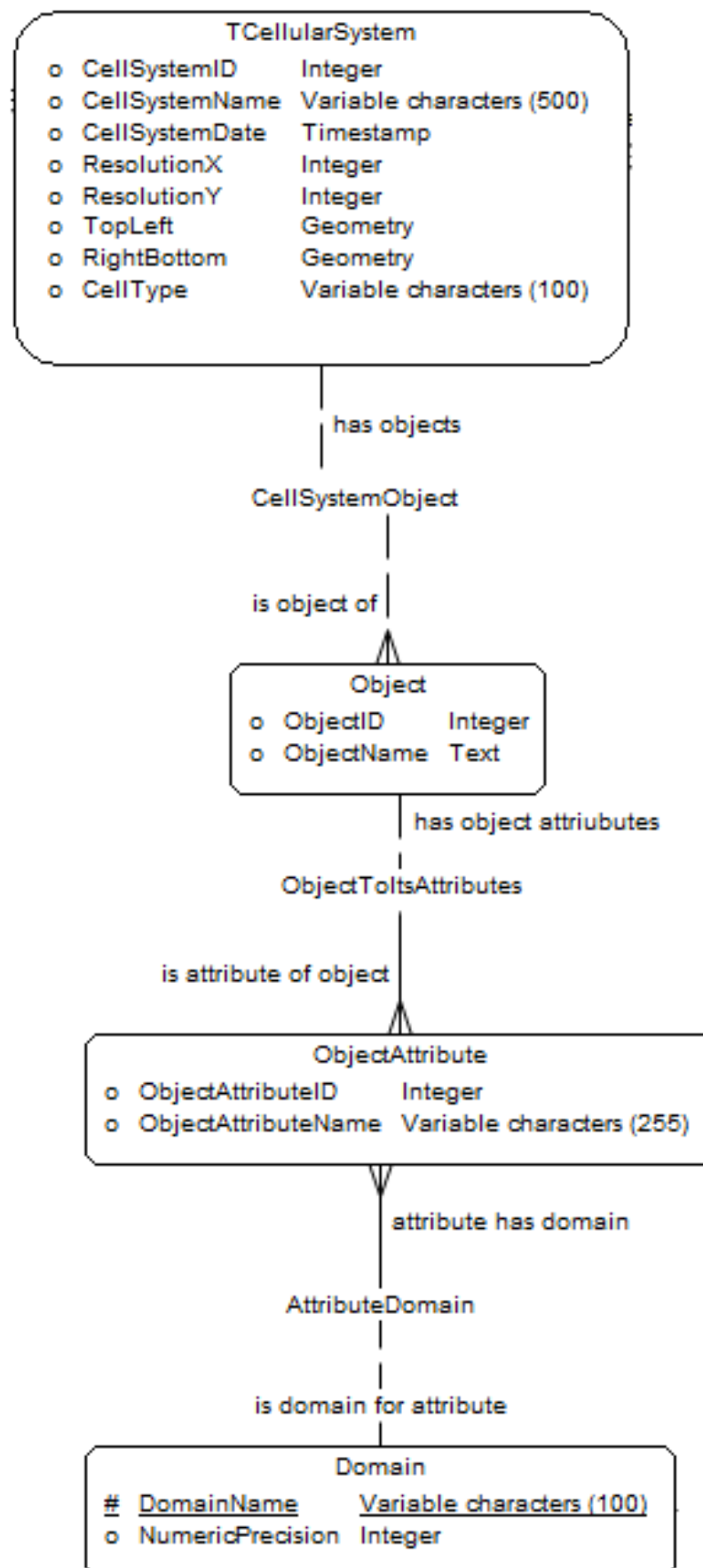


Figure D.1: Descriptions of data relating to CPSs: meteorological data.



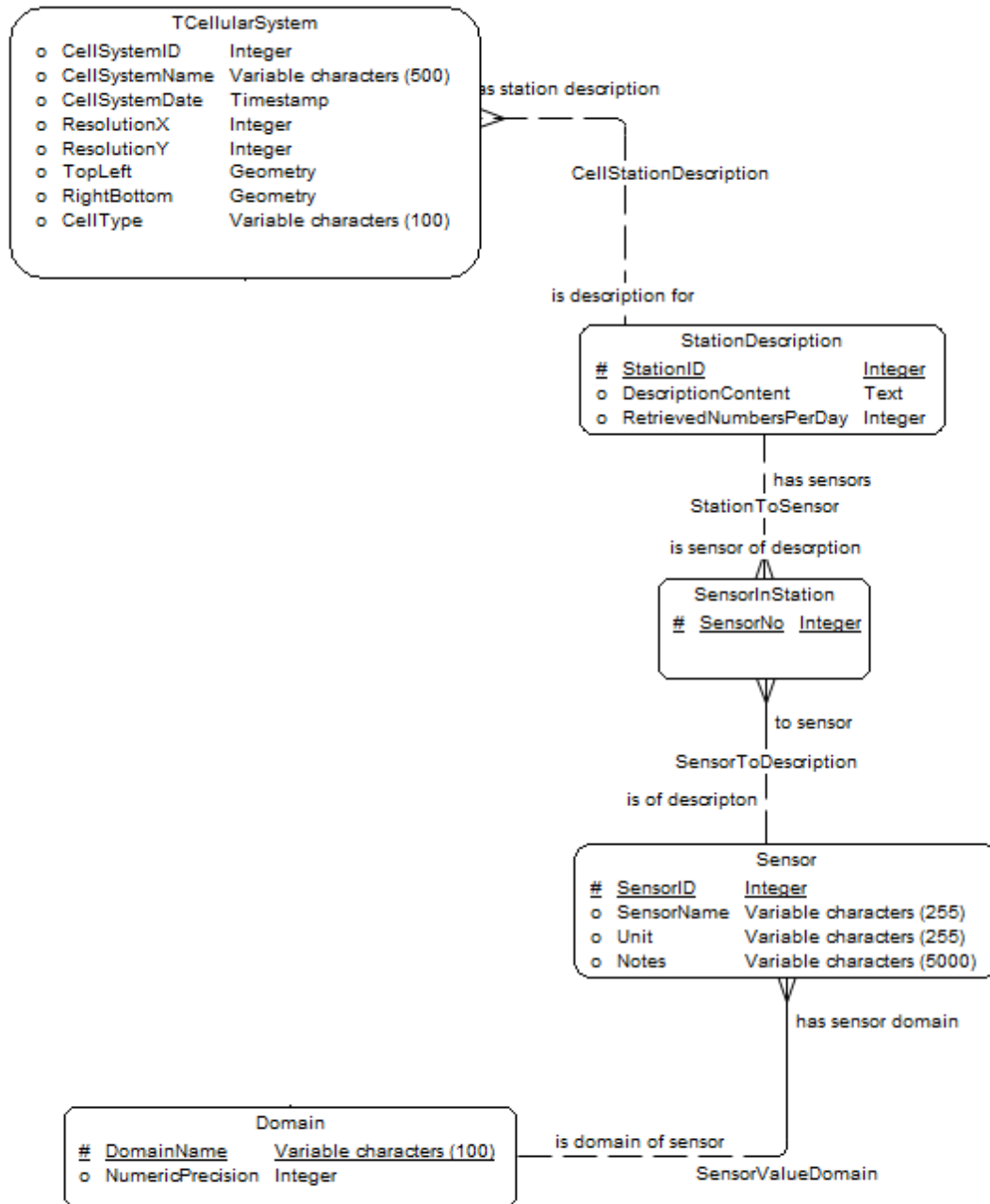


Figure D.3: Descriptions of data relating to CPSs: sensor stations.

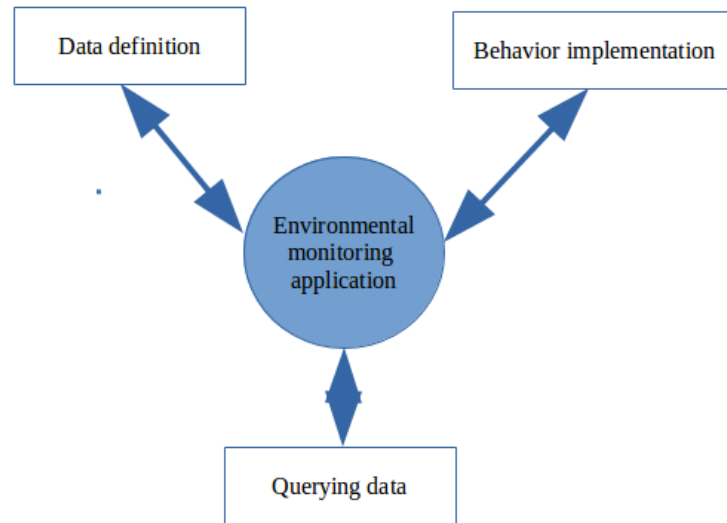


Figure D.4: A proposed workflow to develop an environmental monitoring application.

```

#define MAX_FANOUT 8

typedef struct s_mapped{
    int node;
    int canal;
    int identity;
}mapped;
typedef struct s_canaux{
    int nbOut;
    int nbIn;
    int nbDyn;
    mapped write [MAX_FANOUT];
    mapped read [MAX_FANOUT];
}canaux;
typedef struct{
    int x;
    int y;
    int range;
} node_param;
  
```

Listing D.1: A cellular network for the insect monitoring application. There are 31311 cells with the usage of Moore neighbors.

```

// Sensor node
typedef struct{
    int x,y;
  
```

```

        double lon , lat ;
        float communicationRange ;
        float BPHdensity ;
    } SensorNode ;
    // Objects (parameters)
typedef struct
    {
        float riceStage ; // rice ages
        // Life circle 28 days , eggs 0–6 days ,
        //     nymphs 7–14 days ; adults : 15–28 days
        float hoppers[LIFE_CIRCLE] ;
    } InternalState ;
    // Meteorological data
typedef struct {
        double ugrd ; // composante u du vent
        double vgrd ; // composante v du vent
        double windDirection ; // wind direction according to CA
        double temperature ;
        double humidity ;
        double precipitation ;
        double pressure ;
    } WeatherData ;
    // Cell state
typedef struct
    {
        int seed ; // Random number
        CellPosition cellPosition ; // position
        InternalState in_State ; // Objects
        WeatherData meteo ; // environment factors
        float neighbors[8][LIFE_CIRCLE] ; // For propagating
    } CellState ;

```

Listing D.2: The data relating to the cellular automaton in listing D.1. Wind, temperature, precipitation, humidity are taken into account for simulating insect invasion.

E

Appendix E

E.1 Algorithms for implementing insect behaviors

In implementation, most of operations confront with one dimensional biological cellular automaton BPH (figure 4.3). Actually, this cellular automaton is an array v for which each value $v[i]$ depicts number of *i-days old* BPH. Thus, propagation can be considered as the movement of a number of *k-days old* BPH at a source cell to *l-days old BPH* at a destination cell.

Left side of figure E.1 illustrates the principle of the movement algorithm from a source to a destination buffer. To BPH, only adult individuals are able to migrate due to surrounding conditions, therefore, the migration is the movement of a number of *i-days old* BPH in the source to *i-days old* BPH in the destination ($\forall i \in [15, 27]$). It is

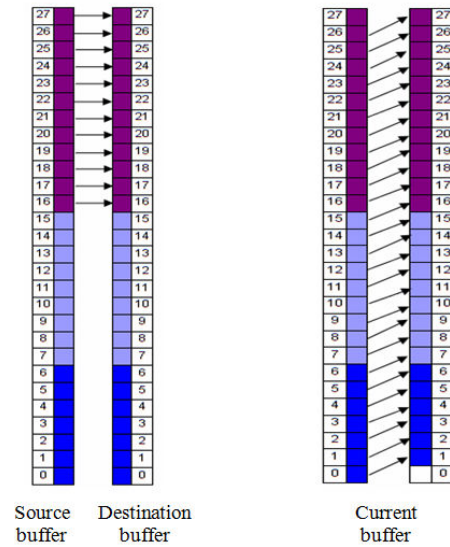


Figure E.1: Implementation of BPH invasion from a source buffer to a destination buffer as well as the self increase in age of BPH after a day.

the principle to implement the function **moveOut** as follows:

```

Input: curCell: the current node
Input: buffCell: the node buffer
Input: j: migration to a destination which is child  $j$  of curCell
Input: pMigrations: percentage of migration
Output: Transfer data from a cell to its buffer
(1)  $density \leftarrow curCell.in\_State.hoppers$ 
(2) for  $i \in [15, LIFE\_CIRCLE - 1]$  do
(3)    $p \leftarrow \_frand(seed, 1.0f, pMigrations)$ 
(4)    $numInsects \leftarrow p/100.0 * density[i]$ 
(5)   if  $numInsects > 0$  then
(6)      $buffCell[j].hoppers[i] \leftarrow buffCell[j].hoppers[i] + numInsects$ 

```

Algorithm 8: BPH migration implementation. Actually, it is the transfer of data from the cell to the cell's buffer. The final transfer will be called in function **readBuffer**.

The right side of figure E.1 illustrates the growth of BPH after a day. The idea is simple: after a day, each specimen has one day old more. Therefore, an i days BPH becomes an $i + 1$ days BPH ($\forall i \in [1..LIFE_CIRCLE - 2]$). Those who are $LIFE_CIRCLE - 1$ days old can be declared dead and number of 0 -day old BPH is 0 for the new day. Details

of the algorithm can be found as follows:

```
Input: curCell: the current node  
Output: Increase each BPH 1 day old  
(1) for  $i \in [1, LIFE\_CIRCLE - 1]$  do  
(2)    $curCell.in\_State.hoppers[i] \leftarrow curCell.in\_State.hoppers[i - 1]$   
(3)  $curCell.in\_State.hoppers[0] \leftarrow 0$   
(4)  $curCell.in\_State.riceStage \leftarrow curCell.in\_State.riceStage + 1$   
    // rice age increase 1
```

Algorithm 9: The self growth algorithm of BPH after day.

E.2 Cells combination into polygon

Algorithm for section 4.3.3 as follows:

Input: Top left and right bottom geographic position (latitude, longitude) of cells which have the same label (infection level) in a text form

Output: A polygon for these cells

```

Begin
pos := position(': ' in str);
Select into ptrResult ST_GeomFromText('GEOMETRYCOLLECTION EMPTY');
ptr := ptrResult;
If (pos!=0) then
s := left(str,pos-1); -- Get the string: the id and the level of polygon
pos1 := position(' ' in s);
-- Get id and level
idS := left(s, pos1-1);
levelS := right(s, character_length(s)-pos1);
-- Process for the rest containing the list of points
str := right(str, character_length(str)-pos);
pos := position('; ' in str);
while (pos!=0) loop
--Get the string for the first points list
s := left(str,pos-1);
-- Get 2 points
pos1 := position(', ' in s);
-- the first point string
tlS := left(s, pos1-1);
-- the second point
rbS := right(s, character_length(s)-pos1);
-- 1st point in the list
pos2 := position('@' in tlS);
x1 := left(tlS, pos2-1);
y1 := right(tlS, character_length(tlS)-pos2);
-- 3rd point in the list
pos3 := position('@' in rbS);
x3 := left(rbS, pos3-1);
y3 := right(rbS, character_length(rbS)-pos3);
polygonStr := 'POLYGON(('||x1||' '||y1||', '||x1||' '||y3||', '
polygonStr := polygonStr ||x3||' '||y3||', '||x3||' '||y1||';
polygonStr := polygonStr || ', ' || x1 || ' ' || y1 || '))';
Select into ptr1 ST_GeomFromText(polygonStr, 4326);
Select into ptrResult ST_Union(ptr, ptr1);
ptr := ptrResult;
-- the rest of the string
str := right(str, character_length(str)-pos);
pos := position('; ' in str);
end loop;
Insert Into ResultPolygon (id, polygonResult, level) Values
(cast(idS as int), ptrResult, cast(levelS as int));
End If;
End;

```

Algorithm 10: Cells combination into a polygon in PostgreSQL code.

Publications

- Bao Hoai Lam, Hiep Xuan Huynh, and Bernard Pottier. **Synchronous networks for bio-environmental surveillance based on cellular automata**. EAI Endorsed Trans. Context-aware Syst. & Appl., 3(8):e5, 2016.
- Bao Hoai Lam, Hiep Xuan Huynh, Mahamadou Traoré, Pierre Yves Lucas, and Bernard Pottier. **Monitoring environmental factors in Mekong Delta of Vietnam using Wireless Sensor Network approach**. In 8th International conference on Simulation and Modelling in the Food and Bio-Industry FoodSim'2014, number ISBN 978-90-77381-84-7, pages 71–78, Brest, France, 2014.
- Bao Hoai Lam, Huu-Hung Huynh, Hiep Xuan Huynh, Bernard Pottier: **A Synchronous Network for Brown Planthopper Surveillance Based on Hexagonal Cellular Automata**. ICTCC 2016: 97-112.
- Bao Hoai Lam, Tuyen Phong Truong, Ky Minh Nguyen, Hiep Xuan Huynh, Bernard Pottier: **An Hierarchical Scheduled Algorithm for Data Dissemination in a Brown Planthopper Surveillance Network**. ICTCC 2016: 246-263.
- Bao Hoai Lam, Hoang Van Tran, Hiep Xuan Huynh, Bernard Pottier: **Synchronous networks for insects surveillance**. SoICT 2015: 163-170.
- Bao Hoai Lam, Tai Tan Phan, Long Huynh Vuong, Hiep Xuan Huynh, Bernard Pottier: **Designing a brown planthoppers surveillance network based on wireless sensor network approach**. CoRR abs/1312.3692 (2013).

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 1.1 | Climate change | 1 |
| 1.1.1 | Impacts | 1 |
| 1.1.2 | Global strategies | 2 |
| 1.1.3 | Past and current international efforts | 3 |
| 1.1.4 | Climate change in Vietnam | 5 |
| 1.2 | Rice insect threat in climate change condition | 7 |
| 1.2.1 | Study of Brown Planthopper threat | 7 |
| 1.2.2 | Manual measurement system: Light trap | 10 |
| 1.2.3 | How to mitigate BPHs threat : Problem statements | 13 |
| 1.3 | System methods to confront BPH | 14 |
| 1.3.1 | Modeling and simulation: A key point | 14 |
| 1.3.2 | Insect classification and counting | 16 |
| 1.3.3 | Simulation and multi-simulation solution | 16 |
| 1.3.4 | Methodology: Sensing, understanding, controlling | 18 |
| 1.3.5 | Thesis layout | 19 |
| 2 | A vision observatory: insect sampling | 21 |
| 2.1 | Automatic observatory - Basic principles | 21 |
| 2.2 | Automatic vision system | 24 |
| 2.2.1 | Objectives of the visual sensor | 24 |
| 2.2.2 | Camera system and technical issues | 25 |
| 2.2.3 | Observation by using fisheye lens camera | 31 |
| 2.3 | Camera calibration in bio-diversity traps | 35 |
| 2.3.1 | Fisheye projection functions | 35 |
| 2.3.2 | Image center calculation | 37 |
| 2.3.3 | Size object estimation | 38 |

| | | |
|----------|--|-----------|
| 2.3.4 | Validation of calibration method | 40 |
| 2.4 | Application of the camera calibration method | 43 |
| 2.4.1 | Linear regression | 43 |
| 2.4.2 | Insect size dataset | 44 |
| 2.4.3 | Insect classification/counting | 45 |
| 2.5 | Experimentation of camera calibration | 46 |
| 2.5.1 | Implementation | 46 |
| 2.5.2 | Results | 46 |
| 2.6 | Operating principals of automatic bio-diversity trap | 49 |
| 2.6.1 | Light traps network | 49 |
| 2.6.2 | Trapping insects | 50 |
| 2.6.3 | Scenarios of trapping | 50 |
| 2.7 | Summary | 51 |
| 3 | Cellular simulation and information system structure | 53 |
| 3.1 | Introduction | 53 |
| 3.2 | Related tools and techniques | 58 |
| 3.2.1 | Modeling and simulation | 58 |
| 3.2.2 | Data and simulation | 60 |
| 3.3 | Cellular automata and sensing machine | 62 |
| 3.3.1 | Sensing station and sensor coverage | 62 |
| 3.3.2 | Coverage problem | 63 |
| 3.3.3 | Maps data and cellular automata modeling | 63 |
| 3.3.4 | Coverage issue modeling | 66 |
| 3.4 | Data access architecture | 67 |
| 3.4.1 | How practical data is accessed | 67 |
| 3.4.2 | Description of data in cellular automata sensing machine | 69 |
| 3.4.3 | Transformation to data definition | 73 |
| 3.5 | Expected behaviors of the environmental machine | 76 |
| 3.5.1 | Growth and propagation algorithms | 76 |
| 3.5.2 | Behavior implementation | 77 |
| 3.5.3 | Some algorithms in spatial CA | 78 |
| 3.6 | Data querying in environmental monitoring applications | 81 |
| 3.6.1 | Wind rose algorithm | 82 |
| 3.6.2 | Data collection from sensor stations | 82 |
| 3.7 | Summary | 84 |

| | | |
|----------|---|------------|
| 4 | Application: Modeling rice insect pest using cellular automata machine | 87 |
| 4.1 | Introduction | 87 |
| 4.2 | Modeling insect invasion using cellular sensing machine | 88 |
| 4.2.1 | Cellular automata synchronous system | 88 |
| 4.2.2 | State variables of cells | 90 |
| 4.2.3 | Data model for insect invasion application | 92 |
| 4.3 | Rice insect pest implementation | 92 |
| 4.3.1 | Retrieval of environmental data from external sources | 92 |
| 4.3.2 | Insect behaviours implementation | 94 |
| 4.3.3 | Simulation result analyse | 96 |
| 4.4 | Experiment | 98 |
| 4.4.1 | Configuration of experimentations | 98 |
| 4.4.2 | Scenario 1: Observing hoppers at a defined location | 100 |
| 4.4.3 | Scenario 2: Hoppers propagation due to wind | 102 |
| 4.5 | Summary | 105 |
| 5 | Conclusions | 107 |
| 5.1 | Thesis results | 107 |
| 5.2 | Local vs. Global Cyber Physical System | 108 |
| 5.3 | NetGen and Pickcell | 109 |
| 5.4 | Future plans | 111 |
| A | Appendix A | 113 |
| A.1 | Validation methods | 113 |
| A.1.1 | Normal distribution | 113 |
| A.1.2 | Shapiro-Wilk test | 114 |
| A.1.3 | Kolmogorov-Smirnov test | 114 |
| A.2 | Validation results | 115 |
| B | Appendix B | 117 |
| B.1 | Bilinear interpolation implementation in PostgreSQL | 117 |
| B.2 | Bilinear interpolation implementation in CUDA | 119 |
| C | Appendix C | 123 |
| C.1 | Sensing machine modeling in NetGen | 123 |

| | |
|--|------------|
| D Appendix D | 127 |
| D.1 Data descriptions for meteorological, object and sensor data | 127 |
| D.2 Workflow for developing an environmental monitoring application . . . | 127 |
| E Appendix E | 133 |
| E.1 Algorithms for implementing insect behaviors | 133 |
| E.2 Cells combination into polygon | 135 |

List of Tables

| | | |
|-----|--|-----|
| 1.1 | Three life cycle of <i>Cyrtorhinus lividipennis</i> Reuters. | 9 |
| 1.2 | Densities of BPHs in light traps collected in Dong Thap, Vietnam from January, 1 st to January, 19 th 2011. Green cells depict the maximum of insects caught in a day of over light traps. | 12 |
| 2.1 | Comparisons between three types of cameras. | 29 |
| 2.2 | Expectation and deviation of measured sizes (pixels and scale) of objects in group 2 (distance around 136.32 pixels from the image center). . . . | 42 |
| 2.3 | Reference table for object sizes in insect trap. | 47 |
| 2.4 | Hopper level related to number of insects caught by the light trap. . . . | 51 |
| 3.1 | Sensing values in a station over the time. | 55 |
| 3.2 | Example of attributes of 2 applications: insect monitoring and radio propagation. | 73 |
| 3.3 | Information of weather factors in an insect monitoring application. . . . | 74 |
| 3.4 | BPHs density and rice ages in insect invasion monitoring application. . . | 74 |
| 4.1 | PostgreSQL and CUDA implementation performance in cells definition phase. | 93 |
| 4.2 | Ascending levels of infested BPHs in rice fields. | 100 |
| A.1 | Expectation and deviation of measured sizes (pixels and scale) of objects in group 1 (distance around 73.32 pixels from the image center). | 116 |

List of Figures

| | | |
|-----|---|----|
| 1.1 | Mekong Delta - A rice region of Vietnam: its rice crop map derived from Envisat ASAR WSM data from 2007 (taken from [123]). | 6 |
| 1.2 | Three stages of development of BPHs. ¹ after hatching | 8 |
| 1.3 | Picture of manual insect light trap taken in O Mon, Can Tho. | 11 |
| 1.4 | Diagram of photacoustic sensor (from [12]) used to recognize insects. | 17 |
| 1.5 | HLA federation architecture used in multi-simulation. | 17 |
| 1.6 | Synchronous points in a federation. | 18 |
| 2.1 | Diagram of local automatic diagnostics showing relation between natural phenomena and sensing devices. | 22 |
| 2.2 | Raspberry 3.0 equipped with a camera. | 25 |
| 2.3 | Jetson Tk1 board with an eCam camera. | 25 |
| 2.4 | Several vision approaches: (a) standard camera separating vision matrix and processing unit, (b) smart sensor with photodiode matrix and processing on the same chip, (c) camera assembly with DMA between the vision matrix and a graphics processing unit memory (GPU). | 26 |
| 2.5 | Camera integrating a parallel processor and sensor array. | 27 |
| 2.6 | The smart camera equipped with a lens. | 27 |
| 2.7 | MAPP220 Block diagram (source from [154]), chip equipped with a photo matrix, 114 digital registers, an analog register and three types of pixel processing units: Point Logical Unit (PLU), Neighborhood Logical Unit (NLU), Global Logical Unit (GLU). | 28 |
| 2.8 | Description of object extraction using MARK operation. In the register, objects vertically connected to objects in the accumulator are preserved (source from [154]). | 29 |
| 2.9 | Pinhole camera with line II' is the optical axis and I' is the principal point or image center. | 30 |

| | | |
|------|---|----|
| 2.10 | Focal length and field of view of a camera lens. | 31 |
| 2.11 | Pictures (view) of an installed automatic bio-diversity trap. | 32 |
| 2.12 | View of cone of bio-diversity trap, on the left an installed device (smart-phone camera) on the right view from the fisheye lens camera | 33 |
| 2.13 | Diagram of a cone surface used in the automatic bio-diversity trap. | 33 |
| 2.14 | Insects caught by the trap, on the left distorted image taken by a fisheye lens camera on the right rectilinear image taken by rectilinear lens camera. | 34 |
| 2.15 | Workflow of insect classification/counting in a bio-diversity trap using fisheye lens camera. | 34 |
| 2.16 | Object size projection in pinhole camera. | 35 |
| 2.17 | Object size projection in fisheye lens camera. | 36 |
| 2.18 | A checkboard pattern which corner points are used to estimate camera parameters. | 37 |
| 2.19 | Diagram of objects located on cone sides of bio-diversity light trap with camera places at O and principal point coincides with the bottom center. | 38 |
| 2.20 | Workflow of camera calibration process used to estimate sample object sizes in a light trap. | 39 |
| 2.21 | Picture of pattern of concentric circle composed by 4mm radius objects used to test distant assumption. | 40 |
| 2.22 | Result of statistical analysis of objects size of group 2 : on left probability density function on right Normalized quantile-quantile plot. | 41 |
| 2.23 | Object pixel scales relate to distance to (the cone) center, the line represents the linear regression. | 43 |
| 2.24 | Distorted view of insect bodies at the surface of bio-diversity light trap cone. | 45 |
| 2.25 | Sample objects detected in the camera calibration. The yellow ellipse depicts the cone bottom. | 47 |
| 2.26 | 2 results of insect counting in a bio-diversity trap. | 48 |
| 2.27 | Regional network composed by three light traps connecting to a center. | 49 |
| 3.1 | Global view of an environmental monitoring application composed by local observatory elements (or nodes) presented in figure 2.1. | 54 |
| 3.2 | Sensors deployment to monitor a service area. Sensing field is a kind of capability to sense environment around the sensor location. | 55 |

| | | |
|------|---|----|
| 3.3 | Snapshot of Brest weather forecast data of September 12 th 2017 (from https://www.worldweatheronline.com/brest-weather/bretagne/fr.aspx). | 56 |
| 3.4 | Meteorological parameters including wind, temperature at Mekong Delta in 04/09/2017 using ZyGrib [http://www.zygrib.org/index.php?page=home] and GFS/NOAA data [https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forecast-system-gfs]. | 56 |
| 3.5 | Simulation for predicting the future of the physical system. | 57 |
| 3.6 | Neighborhood structure for 2 dimensional CA, from left to right: 5 neighbors square (Van Neumann scheme), 9 neighbors square (Moore scheme) and 7 neighbors hexagonal. | 59 |
| 3.7 | Coverage of sensing devices in a geographic area. On left ideal case homogeneous sensing range r_{s_i} , on right more realistic case complex sensing field $f(s_i)$ | 62 |
| 3.8 | Map division into cells to compose a spatial cellular automaton structure, case of Brest harbor with 7 on 4 cells (grid size = 5733m x 5733m). | 64 |
| 3.9 | Sketch structure of a synchronous observation system related to CA: cells or process run same program at a time (parallelism and locality are described). | 65 |
| 3.10 | Structure of a complex multi-layer cellular automata machine at two consecutive states (t and t+1) : transition rules (MG and ST functions) described the state change. | 66 |
| 3.11 | Architecture of data access in environmental monitoring systems. | 68 |
| 3.12 | Representation of sensor stations distributed on study area to sample physical parameters (map with grid size = 3822m x 3822m). | 69 |
| 3.13 | Interaction of data in environmental monitoring systems. | 70 |
| 3.14 | Two CA layers overlapped together, first layer stores wind vector (u, v) data in each cell and second one is holding insect density. | 70 |
| 3.15 | Description of data and object interactions used in environment machines, composed for meteorological data, object and sensor (more details in appendix D.1). | 72 |
| 3.16 | Data model of the BPH invasion automatically generated from data description (figure 3.15). | 75 |

| | | |
|------|--|-----|
| 3.17 | Description of message sending in cellular automata network: process P1 sends a message via channel C to process P2 (from [40]). It is generated from the tool PickCell. | 77 |
| 3.18 | Schema showing how the execution of a CUDA application. The CPU sends data to GPU to manipulate and receives data results after processing for later purposes. | 78 |
| 3.19 | Description of bilinear interpolation at point I based on its four neighbors A_1, A_2, A_3, A_4 | 79 |
| 3.20 | Distance between two points P, Q on a sphere of center O. α_1, ϕ_1 are the longitudinal and latitudinal angles of P, respectively. $\Delta\sigma$ is the angle between the two segments OP and OQ. | 80 |
| 3.21 | Velocity wind vector components u and v in meteorological wind direction. | 81 |
| 3.22 | Wind rose computed at a point of insect invasion application after executing the simulation. | 84 |
| 3.23 | Example of accessing to values of the sensing station 10 from map, values include temperature, wind, humidity, BPHs and not BPHs densities (Map with grid size = 3822m x 3822m). | 85 |
| 4.1 | Map representing insect behaviors in their working space divided in as cells (grid size = 7644m x 7644m). | 89 |
| 4.2 | Description of CA modeling interactions. | 90 |
| 4.3 | Biological cellular automaton illustrating life cycle of BPH in cell based on its age in days. | 91 |
| 4.4 | Cellular system with (40 pixels, 40 pixels) grid, approximately 100km ² per cell, blue arrows representing the wind direction. | 93 |
| 4.5 | Neighbors of a process under wind condition. | 94 |
| 4.6 | Meshed map showing result of the BPH invasion simulation in one step process (grid size = 7644m x 7644m). | 97 |
| 4.7 | Simulation result on a grid represents five distinct polygons a.k.a. area, two of light infection (in blue) 6 and 5 cells, one area of medium infection (in red) and two heavy infection polygons of 3 and 5 cells (in red). | 97 |
| 4.8 | CCL problem represented as topology issue, used to find number of connected vertices. | 98 |
| 4.9 | Example of Union Find algorithm used to solve CCL of figure 3.19. | 99 |
| 4.10 | Light infections at some communes in Can Tho city. | 101 |

| | | |
|------|---|-----|
| 4.11 | Hoppers infection level in Can Tho city for days 3, 4, 6, 7 at several locations. | 101 |
| 4.12 | Map of Hoppers infection in Can Tho city at day 7. | 102 |
| 4.13 | Hoppers infestation in Thoi Lai and Phong Dien in 7 days. | 103 |
| 4.14 | Hoppers infection in Thoi Lai and Phong Dien at day 3. | 104 |
| 4.15 | Map of leeward communes in Phong Dien district, arrow represents the average wind direction | 104 |
| 5.1 | Concept map of Cyber Physical System ² | 110 |
| A.1 | Probability density function of a normal distribution. | 114 |
| A.2 | Some statistical results of objects in group 1: Probability density function, quantile-quantile (q-q) plot of object scales in the image. | 115 |
| C.1 | Work flow for modeling insect physical and WSN system. | 124 |
| D.1 | Descriptions of data relating to CPSs: meteorological data. | 128 |
| D.2 | Descriptions of data relating to CPSs: object data. | 129 |
| D.3 | Descriptions of data relating to CPSs: sensor stations. | 130 |
| D.4 | A proposed workflow to develop an environmental monitoring application. | 131 |
| E.1 | Implementation of BPH invasion from a source buffer to a destination buffer as well as the self increase in age of BPH after a day. | 134 |

Bibliography

- [1] Agence France Presse (2013). Locust plague threatens to trigger severe food crisis in Madagascar (in FRENCH). <http://www.fao.org/emergencies/resources/videos/video-detail/en/c/178727/>.
- [2] Alliance, Z. (2017). Zigbee ® technology. <http://www.zigbee.org/>.
- [3] Ando, B., Baglio, S., and Trigona, C. (2010). Autonomous sensors: From standard to advanced solutions [instrumentation notes]. *IEEE Instrumentation Measurement Magazine*, 13(3):33–37, ISSN: 1094–6969, DOI: 10.1109/MIM.2010.5475165.
- [4] Ariel, G. and Ayali, A. (2015). Locust collective motion and its modeling. *PLOS Computational Biology*, 11(12):1–25, DOI: 10.1371/journal.pcbi.1004522, <https://doi.org/10.1371/journal.pcbi.1004522>.
- [5] Arnett, E. B., Hein, C. D., Schirmacher, M. R., Huso, M. M. P., and Szewczak, J. M. (2013). Evaluating the effectiveness of an ultrasonic acoustic deterrent for reducing bat fatalities at wind turbines. In *PloS one*.
- [6] Åström, A. and Forchheimer, R. (1992). MAPP2200 smart vision sensor. programmability and adaptivity. In [174], pages 17–20, <http://b2.cvl.iis.u-tokyo.ac.jp/mva/proceedings/CommemorativeDVD/1992/papers/1992017.pdf>.
- [7] Atanbori, J., Duan, W., Murray, J., Appiah, K., and Dickinson, P. (2015). Automatic classification of flying bird species using computer vision techniques. *Pattern Recognition Letters*, pages –, ISSN: 0167–8655, DOI: <http://dx.doi.org/10.1016/j.patrec.2015.08.015>, <http://www.sciencedirect.com/science/article/pii/S0167865515002743>.
- [8] Balch, T., Khan, Z., and Veloso, M. (2001). Automatically tracking and analyzing the behavior of live insect colonies. In *Proceedings of the Fifth International Conference on Autonomous Agents*, AGENTS '01, pages 521–528, New

- York, NY, USA. ACM, ISBN: 1-58113-326-X, DOI: 10.1145/375735.376434, <http://doi.acm.org/10.1145/375735.376434>.
- [9] Balis, B., Bubak, M., Harezlak, D., Nowakowski, P., Pawlik, M., and Wilk, B. (2017). Towards an operational database for real-time environmental monitoring and early warning systems. *Procedia Computer Science*, 108:2250 – 2259, ISSN: 1877-0509, DOI: <http://dx.doi.org/10.1016/j.procs.2017.05.193>, <http://www.sciencedirect.com/science/article/pii/S1877050917307846>. International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland.
- [10] Barnaghi, P., Wang, W., Henson, C., and Taylor, K. (2012). Semantics for the Internet of Things: Early Progress and Back to the Future. *Int. J. Semant. Web Inf. Syst.*, 8(1):1–21, ISSN: 1552-6283, DOI: 10.4018/jswis.2012010101, <http://dx.doi.org/10.4018/jswis.2012010101>.
- [11] Barrett, Jr., J. R., Harwood, F. W., and Deay, H. O. (1972). Functional association of light trap catches to emission of blacklight fluorescent lamps 1. *Environmental Entomology*, 1(3):285–290, DOI: 10.1093/ee/1.3.285, [+http://dx.doi.org/10.1093/ee/1.3.285](http://dx.doi.org/10.1093/ee/1.3.285).
- [12] Batista, G. E., Keogh, E. J., Mafra-Neto, A., and Rowton, E. (2011). Sigkdd demo: Sensors and software to allow computational entomology, an emerging application of data mining. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11*, pages 761–764, New York, NY, USA. ACM, ISBN: 978-1-4503-0813-7, DOI: 10.1145/2020408.2020530, <http://doi.acm.org/10.1145/2020408.2020530>.
- [13] Bengio, Y. (2009). Learning Deep Architectures for AI. *Foundations and Trends @in Machine Learning*, 2(1):1–127, ISSN: 1935-8237, DOI: 10.1561/22000000006, <http://dx.doi.org/10.1561/22000000006>.
- [14] Bergström, L., Delsing, P., L’Huillier, A., and Inganäs, O. (2014). Blue LEDs - Filling the world with new light. https://www.nobelprize.org/nobel_prizes/physics/laureates/2014/popular-physicsprize2014.pdf. The Nobel Prize in Physics 2014 - The Royal Swedish Academy of Sciences.

- [15] Bernard Pottier and Pierre-Yves Lucas (2015). Dynamic networks "Netgen: objectives, installation, use, and programming". <https://github.com/NetGenProject>. Université de Bretagne Occidentale.
- [16] Beyan, C. (2016). *Fish Behavior Analysis*, pages 161–179. Springer International Publishing, Cham, ISBN: 978-3-319-30208-9, DOI: 10.1007/978-3-319-30208-9_12, http://dx.doi.org/10.1007/978-3-319-30208-9_12.
- [17] Bowden, J. (1981). The relationship between light- and suction-trap catches of *chrysoperla carnea* (stephens) (neuroptera: Chrysopidae), and the adjustment of light-trap catches to allow for variation in moonlight. *Bulletin of Entomological Research*, 71(4):621–629, DOI: 10.1017/S0007485300010142.
- [18] Bowden, J. (1982). An analysis of factors affecting catches of insects in light-traps. *Bulletin of Entomological Research*, 72:535–556, ISSN: 1475-2670, DOI: 10.1017/S0007485300008579, http://journals.cambridge.org/article_S0007485300008579.
- [19] Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.
- [20] Briscoe, A. D. and Chittka, L. (2001). The evolution of color vision in insects. *Annual review of entomology*, 46:471–510.
- [21] Broggi, A., Cerri, P., Grisleri, P., and Paterlini, M. (2009). *Boat Speed Monitoring Using Artificial Vision*. Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-04146-4, DOI: 10.1007/978-3-642-04146-4_36, http://dx.doi.org/10.1007/978-3-642-04146-4_36.
- [22] Budkowski, S. and Dembinski, P. (1987). An introduction to Estelle: A Specification Language for Distributed Systems. *Comput. Netw. ISDN Syst.*, 14(1):3–23, ISSN: 0169-7552, DOI: 10.1016/0169-7552(87)90084-5, [http://dx.doi.org/10.1016/0169-7552\(87\)90084-5](http://dx.doi.org/10.1016/0169-7552(87)90084-5).
- [23] Cabauatan, P. Q., Cabunagan, R. C., and Choi, I.-R. (2009). Rice viruses transmitted by the brown planthopper *Nilaparvata lugens* Stål. In Heong, K. L. and Hardy, B., editors, *Planthoppers: new threats to the sustainability of intensive rice production*

- systems in Asia*, pages 357–368. International Rice Research Institute, Asian Development Bank, Australian Government, Australian Centre for International Agricultural Research. ISBN: 9789712202513.
- [24] Carlo Nardone (2015). Cuda programming model. <http://www.sie.es/wp-content/uploads/2015/12/nvidia-cuda-programming-model.pdf>.
- [25] Catindig, L., Arida, G., Baehaki, S., Bentur, J., Cuong, L., Norowi, M., Rattanakarn, W., Sriratanasak, W., Xia, J., and Lu, Z. (2009). Situation of planthoppers in Asia. In Heong, K. L. and Hardy, B., editors, *Planthoppers: new threats to the sustainability of intensive rice production systems in Asia*, pages 191–220. International Rice Research Institute, Asian Development Bank, Australian Government, Australian Centre for International Agricultural Research. ISBN: 9789712202513.
- [26] Chakrabarty, K., Iyengar, S. S., Qi, H., and Cho, E. (2002). Grid coverage for surveillance and target location in distributed sensor networks. *IEEE Trans. Comput.*, 51(12):1448–1453, ISSN: 0018-9340, DOI: 10.1109/TC.2002.1146711, <http://dx.doi.org/10.1109/TC.2002.1146711>.
- [27] Chakravarti, I. M., Laha, R. G., and Roy, J. (1967). *Handbook of Methods of Applied Statistics*, volume I. John Wiley and Sons, USA.
- [28] Chakravarti, I. M., Laha, R. G., and Roy, J. (1987). *Cartographical Innovations: an International Handbook of Mapping Terms to 1900*. Tring, Herts: Map Collector Publications in association with International Cartographic Association.
- [29] Chattopadhyay, D. and Rakshit, P. (2004). *Elements of Physics (Third Edition)*, volume 2. New Age International (P) Limited. ISBN 10: 8122415466 / 13: 9788122415469.
- [30] Chen, J., Hill, A. A., and Urbano, L. D. (2009). A GIS-based model for urban flood inundation. *Journal of Hydrology*, 373(1):184 – 192, ISSN: 0022-1694, DOI: <https://doi.org/10.1016/j.jhydrol.2009.04.021>, <http://www.sciencedirect.com/science/article/pii/S0022169409002546>.
- [31] Cheng, S., Chen, J., Si, H., Yan, L., Chu, T., Wu, C., Chien, J., and Yan, C. (1979). Studies on the migrations of brown planthoppers *nilaparvata lugens*. *Std. Acta Entomol. Sinica*, 22:1–21. In Chinese, English summary.

- [32] Chien H. V., Cuong L. Q., Van D. V., Heong K. L., Cabunagan R. C., and Choi I.R. (2012). Management of rice planthopper and related virus diseases in mekong delta, south vietnam. In *Proceedings of the NARO International Symposium 2012, New Insight into Insecticide Resistance of Rice Planthoppers and their Associated Virus Diseases*. Fukuoka: NARO Kyushu Okinawa Agricultural Research Center.
- [33] Choi, J., Dumortier, Y., Choi, S. I., Ahmad, M. B., and Medioni, G. (2012). Real-time 3-d face tracking and modeling from aw webcam. In *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*, pages 33–40. ISSN: 1550-5790, DOI: 10.1109/WACV.2012.6163031.
- [34] Chuang, M. C., Hwang, J. N., Williams, K., and Towler, R. (2015). Tracking live fish from low-contrast and low-frame-rate stereo videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(1):167–179, ISSN: 1051-8215.
- [35] Climate Change, I. P. (2007). *Fourth Assessment Report: Climate Change 2007: The AR4 Synthesis Report*. Geneva: IPCC, <http://www.ipcc.ch/ipccreports/ar4-wg1.htm>.
- [36] Conrad, K. F., Fox, R., and Woiwod, I. P. (2007). Monitoring biodiversity: measuring long-term changes in insect abundance. In Stewart, A. J. A., New, T. R., and Lewis, O. T., editors, *Insect conservation biology*, pages 203–225. Cabi Publisher. ISBN: 9781845932541.
- [37] Council, N. R. (2010). *Advancing the Science of Climate Change*. The National Academies Press, Washington, DC, ISBN: 978-0-309-14588-6, DOI: 10.17226/12782, <https://www.nap.edu/catalog/12782/advancing-the-science-of-climate-change>.
- [38] Cronin, J. T. (2003). Movement and spatial population structure of a prairie planthopper. *Ecology*, 84(5):1179–1188, ISSN: 1939-9170, DOI: 10.1890/0012-9658(2003)084[1179:MASPSO]2.0.CO;2, [http://dx.doi.org/10.1890/0012-9658\(2003\)084\[1179:MASPSO\]2.0.CO;2](http://dx.doi.org/10.1890/0012-9658(2003)084[1179:MASPSO]2.0.CO;2).
- [39] Danh, V. T. and Mushtaq, S. (2011). *Living with Floods: An Evaluation of the Resettlement Program of the Mekong Delta of Vietnam*, pages 181–204. Springer Netherlands, Dordrecht, ISBN: 978-94-007-0934-8, DOI: 10.1007/978-94-007-0934-8_11, https://doi.org/10.1007/978-94-007-0934-8_11.

- [40] Daniel C. Hyde (1995). Introduction to the Programming Language Occam. <https://www.eg.bucknell.edu/~cs366/occam.pdf>.
- [41] Ding, W. and Taylor, G. (2016). Automatic moth detection from trap images for pest management. *Computers and Electronics in Agriculture*, 123:17 – 28, ISSN: 0168-1699, DOI: <http://dx.doi.org/10.1016/j.compag.2016.02.003>, <http://www.sciencedirect.com/science/article/pii/S0168169916300266>.
- [42] Dixon, J., Gulliver, A., and Gibbon, D. (2001). *Farming Systems and Poverty: Improving Farmers' Livelihoods in a Changing World*. FAO, Rome.
- [43] Dufay, C. (1964). Contribution à l'étude du phototropisme des lépidoptères Noctuides. *Annales des Sciences Naturelles Zoologie, Paris*. 12. 6 281-406.
- [44] Edelstein-Keshet, L., Watmough, J., and Grunbaum, D. (1998). Do traveling band solutions describe cohesive swarms? An investigation for migratory locusts. *Journal of Mathematical Biology*, 36(6):515–549, ISSN: 1432-1416, DOI: 10.1007/s002850050112, <https://doi.org/10.1007/s002850050112>.
- [45] Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, ISSN: 0018-9162, DOI: 10.1109/2.30720, <http://dx.doi.org/10.1109/2.30720>.
- [46] Esker, P., Obrycki, J., and Nutter, Jr, F. (2004). Trap Height and Orientation of Yellow Sticky Traps Affect Capture of *Chaetocnema pulicaria* (Coleoptera: Chrysomelidae). 97:145–9.
- [47] Estivill-Castro, V., Lattin, D., Suraweera, F., and Vithanage, V. (2003). Tracking bees - a 3d, outdoor small object environment. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 3, pages III–1021–4 vol.2. ISSN: 1522-4880, DOI: 10.1109/ICIP.2003.1247421.
- [48] Fabregat, G., Leon, G., Le Berre, O., and Pottier, B. (1999). Embedded system modeling and synthesis in OO environments. A smart-sensor case study. *Compiler and Architecture Support for Embedded Systems (CASES'99)*.
- [49] Feldhoff, J. H., Lange, S., Volkholz, J., Donges, J. F., Kurths, J., and Gerstengarbe, F.-W. (2015). Complex networks for climate model evaluation with application to statistical versus dynamical modeling of south american climate. *Climate Dynamics*,

- 44(5):1567–1581, ISSN: 1432-0894, DOI: 10.1007/s00382-014-2182-9, <https://doi.org/10.1007/s00382-014-2182-9>.
- [50] Feller, W. (1968). *An Introduction to Probability Theory, Vol. 1*. Wiley, New York, NY, third edition.
- [51] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, ISSN: 0001-0782, DOI: 10.1145/358669.358692, <http://doi.acm.org/10.1145/358669.358692>.
- [52] Fisher, B., Nakicenovic, N., Alfsen, K., Corfee-Morlot, J., and Riahi, K. (2007). Issues related to mitigation in the long-term context (chapter 3). In Metz, B., Davidson, O., Bosch, P., Dave, R., and Meyer, L., editors, *Climate Change 2007: Mitigation. Contribution of WG III to the Fourth Assessment Report of the IPCC*. Cambridge University Press, Cambridge, <http://pure.iiasa.ac.at/8248/>.
- [53] Food and Agriculture Organization of the United Nations (2013). Post Disaster Needs Assessment (PDNA) - Volume A. http://www.recoveryplatform.org/assets/publication/PDNA/PDNA_Global_Facility_for_Disaster_Reduction_and_Recovery.
- [54] Food and Agriculture Organization of the United Nations (2014). Viet Nam. <http://www.fao.org/faostat/en/#country/237>.
- [55] Food and Agriculture Organization of the United Nations (2015). The Impact of Natural Hazards and Disasters on Agriculture and Food Security and Nutrition. <http://www.fao.org/3/a-i4434e.pdf>. A call for action to build resilient livelihoods.
- [56] Food and Agriculture Organization of the United Nations (2016). FAO’s work of climate change. <http://www.fao.org/3/a-i6273e.pdf>. United Nations Climate Change Conference.
- [57] Forum, M. P. (1994). MPI: A Message-Passing Interface Standard. Technical report, Knoxville, TN, USA.
- [58] Francesco Berto and Jacopo Tagliabue (26 Mars 2012). Cellular automata. <http://plato.stanford.edu/entries/cellular-automata/>.

- [59] Freedman, D. (2005). *Statistical Models : Theory and Practice*. Cambridge University Press, ISBN: 0521854830.
- [60] Friis, H. T. (1946). A note on a simple transmission formula. *Proceedings of the IRE*, 34(5):254–256, ISSN: 0096-8390, DOI: 10.1109/JRPROC.1946.234568.
- [61] Galil, Z. and Italiano, G. F. (1991). Data structures and algorithms for disjoint set union problems. *ACM Comput. Surv.*, 23(3):319–344, ISSN: 0360-0300, DOI: 10.1145/116873.116878, <http://doi.acm.org/10.1145/116873.116878>.
- [62] Gao, D., Liu, D., Feng, Y., An, Q., and Yu, F. (2010). *A Robust Image Transmission Scheme for Wireless Channels Based on Compressive Sensing*, pages 334–341. Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-14932-0, DOI: 10.1007/978-3-642-14932-0_42, https://doi.org/10.1007/978-3-642-14932-0_42.
- [63] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645 – 1660, ISSN: 0167-739X, DOI: <http://dx.doi.org/10.1016/j.future.2013.01.010>, <http://www.sciencedirect.com/science/article/pii/S0167739X13000241>. Including Special sections: Cyber-enabled Distributed Computing for Ubiquitous Cloud and Network Services & Cloud Computing and Scientific Applications - Big Data, Scalable Analytics, and Beyond.
- [64] Guckenheimer, J. and Holmes, P. (1983). *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields / John Guckenheimer, Philip Holmes*. Springer-Verlag New York, ISBN: 0387908196 3540908196, <http://www.loc.gov/catdir/enhancements/fy0812/82019641-t.html>.
- [65] Hart, N. H. and Huang, L. (2012). Counting insects in flight using image processing techniques. In *Proceedings of the 27th Conference on Image and Vision Computing New Zealand, IVCNZ '12*, pages 274–278, New York, NY, USA. ACM, ISBN: 978-1-4503-1473-2, DOI: 10.1145/2425836.2425891, <http://doi.acm.org/10.1145/2425836.2425891>.
- [66] Hartley, R. and Kang, S. B. (2007). Parameter-free radial distortion correction with center of distortion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1309–1321, ISSN: 0162-8828, DOI: 10.1109/TPAMI.2007.1147.

- [67] Hassan, U., Bassora, M., Vahid, A. H., O’Riain, S., and Curry, E. (2013). A collaborative approach for metadata management for internet of things: Linking micro tasks with physical objects. In *9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 593–598. DOI: 10.4108/icst.collaboratecom.2013.254174.
- [68] Hawick, K., Leist, A., and Playne, D. (2010). Parallel graph component labelling with GPUs and CUDA. *Parallel Computing*, 36(12):655 – 678, ISSN: 0167-8191, DOI: <https://doi.org/10.1016/j.parco.2010.07.002>, <http://www.sciencedirect.com/science/article/pii/S0167819110001055>.
- [69] Herselman, P. L. and de Wind, H. J. (2008). Improved covariance matrix estimation in spectrally inhomogeneous sea clutter with application to adaptive small boat detection. In *2008 International Conference on Radar*, pages 94–99. ISSN: 1097-5764, DOI: 10.1109/RADAR.2008.4653898.
- [70] Hillis, W. D. and Steele, Jr., G. L. (1986). Data parallel algorithms. *Commun. ACM*, 29(12):1170–1183, ISSN: 0001-0782, DOI: 10.1145/7902.7903, <http://doi.acm.org/10.1145/7902.7903>.
- [71] Ho Van Chien, Nguyen Huu Huan, and Le Quoc Cuong (2007). Rice "Escapes" Brown Plant Hoppers in Mekong Delta. <http://www.fao.org/asiapacific/news/detail-events/en/c/46139/>.
- [72] Hoare, C. A. R. (1978). Communicating sequential processes. *Commun. ACM*, 21(8):666–677, ISSN: 0001-0782, DOI: 10.1145/359576.359585, <http://doi.acm.org/10.1145/359576.359585>.
- [73] Hopcroft, J. and Ullman, J. (1979). *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley Publishing Company.
- [74] Horenstein, H. (2005). *Black & White Photography: A Basic Manual*. Little, Brown and Company, Time Warner Book Group. ISBN 0-316-37305-2.
- [75] Huang, C.-F. and Tseng, Y.-C. (2003). The coverage problem in a wireless sensor network. In *Proceedings of the 2Nd ACM International Conference on Wireless Sensor Networks and Applications*, WSNA '03, pages 115–121, New York, NY, USA. ACM, ISBN: 1-58113-764-8, DOI: 10.1145/941350.941367, <http://doi.acm.org/10.1145/941350.941367>.

- [76] Hughes, C., Denny, P., Jones, E., and Glavin, M. (2010). Accuracy of fish-eye lens models. *Applied Optics*, 49:3338, DOI: 10.1364/AO.49.003338.
- [77] Hughes, C., McFeely, R., Denny, P., Glavin, M., and Jones, E. (2010). Equidistant ($f(\theta)$) Fish-eye Perspective with Application in Distortion Centre Estimation. *Image Vision Comput.*, 28(3):538–551, ISSN: 0262-8856, DOI: 10.1016/j.imavis.2009.09.001, <http://dx.doi.org/10.1016/j.imavis.2009.09.001>.
- [78] Huynh, V. K., Zucker, J. D., Huynh, H. X., and Drogoul, A. (2012). Spatial estimator of brown plant hopper density from light traps data. In *2012 IEEE RIVF International Conference on Computing Communication Technologies, Research, Innovation, and Vision for the Future*, pages 1–6. DOI: 10.1109/rivf.2012.6169850.
- [79] IEEE (2010a). IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)– Federate Interface Specification. *IEEE Std 1516.1-2010 (Revision of IEEE Std 1516.1-2000)*, pages 1–378, DOI: 10.1109/IEEESTD.2010.5557728.
- [80] IEEE (2010b). IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)– Object Model Template (OMT) Specification. *IEEE Std 1516.2-2010 (Revision of IEEE Std 1516.2-2000)*, pages 1–110, DOI: 10.1109/IEEESTD.2010.5557731.
- [81] International Rice Research Institute (1967). *The Major Insect Pests of the Rice Plant: Proceedings*. International Rice Research Institute, <https://books.google.fr/books?id=JNInAQAAMAAJ>. Chapter 5 page 169–182.
- [82] International Rice Research Institute (1979). *Brown Planthopper: Threat to Rice Production in Asia*. Int. Rice Res. Inst, ISBN: 9711040220.
- [83] Itami., R. M. (1994). Simulating spatial dynamics: cellular automata theory. *Landscape and Urban Planning* 30., pages 27–47.
- [84] Jedeliza B. Ferrater (2015). *Adaptation of the brown planthopper, Nilaparvata lugens (Stål), to resistant rice varieties*. PhD thesis, Wageningen University, Wageningen, NL. ISBN 978-94-6257-559-2.
- [85] Ji W and Mitchell L C (2004). Analytical model-based decision support GIS for wetland resource management. In Lyon J G, M. J., editor, *Wetland and environmental applications of GIS*, pages 31–45. Lewis Publishers, New York.

- [86] Jonason, D., Franzén, M., and Ranius, T. (2014). Surveying Moths Using Light Traps: Effects of Weather and Time of Year. *PLoS ONE*, 9:e92453, DOI: 10.1371/journal.pone.0092453.
- [87] Jones, P. G. and Thornton, P. K. (2009). Croppers to livestock keepers: livelihood transitions to 2050 in africa due to climate change. *Environmental Science & Policy*, 12(4):427–437.
- [88] Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., Iredell, M., Saha, S., White, G., Woollen, J., Zhu, Y., Leetmaa, A., Reynolds, R., Chelliah, M., Ebisuzaki, W., Higgins, W., Janowiak, J., Mo, K. C., Ropelewski, C., Wang, J., Jenne, R., and Joseph, D. (1996). The NCEP/NCAR 40-year re-analysis project. *Bulletin of the American Meteorological Society*, 77(3):437–471, DOI: 10.1175/1520-0477(1996)077<0437:TNYRP>2.0.CO;2, [https://doi.org/10.1175/1520-0477\(1996\)077<0437:TNYRP>2.0.CO;2](https://doi.org/10.1175/1520-0477(1996)077<0437:TNYRP>2.0.CO;2).
- [89] Kashiha, M. A., Green, A. R., Sales, T. G., Bahr, C., Berckmans, D., and Gates, R. S. (2014). Performance of an image analysis processing system for hen tracking in an environmental preference chamber. *Poultry Science*, DOI: 10.3382/ps.2014-04078, <http://ps.oxfordjournals.org/content/early/2014/07/26/ps.2014-04078.abstract>.
- [90] Kawahara, A. Y. and Barber, J. R. (2015). Tempo and mode of antitubat ultrasound production and sonar jamming in the diverse hawkmoth radiation. *Proceedings of the National Academy of Sciences*, 112(20):6407–6412, DOI: 10.1073/pnas.1416679112, <http://www.pnas.org/content/112/20/6407.abstract>.
- [91] Kayama, Y. (2010). Complex networks derived from cellular automata. *CoRR*, abs/1009.4509.
- [92] Keith Cressman, Alice Van der Elstraeten, and Clare Pedrick (2016). eLocust3: An innovative tool for crop pest control. <http://www.fao.org/3/a-i6058e.pdf>.
- [93] Keller, K., Steffe, A. S., Lowry, M., Murphy, J. J., and Suthers, I. M. (2016). Monitoring boat-based recreational fishing effort at a nearshore artificial reef with a shore-based camera. *Fisheries Research*, 181:84 – 92, ISSN: 0165-7836, DOI: <http://dx.doi.org/10.1016/j.fishres.2016.03.025>, <http://www.sciencedirect.com/science/article/pii/S0165783616300935>.

- [94] Kenneth C. Laudon, C. (2011.). *Management Information Systems (12th Edition)*. Prentice Hall.
- [95] K.Hejbudzka, R.Lindenbergh, and Soudarissanane, S. (2010). Influence of atmospheric conditions on the range distance and number of returned points in leica scanstation 2 point clouds.
- [96] Kim, Y., Glenn, D. M., Park, J., Ngugi, H. K., and Lehman, B. L. (2010). Hyper-spectral image analysis for plant stress detection. In *American Society of Agricultural and Biological Engineers*. doi:10.13031/2013.29814.
- [97] Kizaki, S. and Katori, M. (1999). A stochastic lattice model for locust outbreak. *Physica A: Statistical Mechanics and its Applications*, 266(1):339 – 342, ISSN: 0378-4371, DOI: [http://dx.doi.org/10.1016/S0378-4371\(98\)00613-X](http://dx.doi.org/10.1016/S0378-4371(98)00613-X), <http://www.sciencedirect.com/science/article/pii/S037843719800613X>.
- [98] Kolmogorov, A. N. (1933). Sulla Determinazione Empirica di una Legge di Distribuzione. *Giornale dell'Istituto Italiano degli Attuari*, 4:83–91.
- [99] Kreft, S., Eckstein, D., and Melchior, I. (2017). Global climate risk index 2017. <https://germanwatch.org/en/download/16411.pdf>.
- [100] Kuhl, F., Weatherly, R., and Dahmann, J. (1999). *Creating Computer Simulation Systems: An Introduction to the High Level Architecture*. Prentice Hall.
- [101] L., N. and J., P. (2011). Efficiency of light traps in relation the number of caught species. *International journal of science and nature*, 2:161–167.
- [102] L Altshuler, D., B Dickson, W., T Vance, J., P Roberts, S., and H Dickinson, M. (2006). Short-amplitude high-frequency wing strokes determine the aerodynamics of honeybee flight. 102:18213–8.
- [103] Lam, B. H., Huynh, H., Huynh, H. X., and Pottier, B. (2016a). A synchronous network for brown planthopper surveillance based on hexagonal cellular automata. In [197], pages 97–112, ISBN: 978-3-319-46908-9, DOI: 10.1007/978-3-319-46909-6_10, http://dx.doi.org/10.1007/978-3-319-46909-6_10.
- [104] Lam, B. H., Huynh, H. X., and Pottier, B. (2016b). Synchronous networks for bio-environmental surveillance based on cellular automata. *EAI Endorsed Trans.*

- Context-aware Syst. & Appl.*, 3(8):e5, DOI: 10.4108/eai.9-3-2016.151117, <http://dx.doi.org/10.4108/eai.9-3-2016.151117>.
- [105] Lam, B. H., Huynh, H. X., Traoré, M., Lucas, P. Y., and Pottier, B. (2014). Monitoring environmental factors in Mekong Delta of Vietnam using Wireless Sensor Network approach. In *8th International conference on Simulation and Modelling in the Food and Bio-Industry FoodSim'2014*, number ISBN 978-90-77381-84-7, pages 71–78, Brest, France.
- [106] Lam, B. H., Phan, T. T., Vuong, L. H., Huynh, H. X., and Pottier, B. (2013a). Designing a brown planthoppers surveillance network based on wireless sensor network approach. *CoRR*, abs/1312.3692, <http://arxiv.org/abs/1312.3692>.
- [107] Lam, B. H., Tran, H. V., Huynh, H. X., and Pottier, B. (2015). Synchronous networks for insects surveillance. In [177], pages 163–170, ISBN: 978-1-4503-3843-1, DOI: 10.1145/2833258.2833295, <http://doi.acm.org/10.1145/2833258.2833295>.
- [108] Lam, B. H., Truong, T. P., Nguyen, K. M., Huynh, H. X., and Pottier, B. (2016c). An hierarchical scheduled algorithm for data dissemination in a brown planthopper surveillance network. In [197], pages 246–263, ISBN: 978-3-319-46908-9, DOI: 10.1007/978-3-319-46909-6_23, http://dx.doi.org/10.1007/978-3-319-46909-6_23.
- [109] Lam, H. B., Phan, T. T., Vuong, L. H., Huynh, H. X., and Pottier, B. (2013b). Designing a brown planthoppers surveillance network based on wireless sensor network approach. In *ISCRAM (Information Systems for Crisis Response and Management) Vietnam 2013 Conference*.
- [110] Larsen, L. (1999). GIS in environmental monitoring and assessment. *Geographical information systems*, 2:999–1007.
- [111] Lasnier, G., Cardoso, J., Siron, P., Pagetti, C., and Derler, P. (2013). Distributed simulation of heterogeneous and real-time systems. In *Distributed Simulation and Real Time Applications (DS-RT), 2013 IEEE/ACM 17th International Symposium on*, pages 55–62. ISSN: 1550-6525, DOI: 10.1109/DS-RT.2013.14.

- [112] Lee, E. A. (2010). CPS Foundations. In *Proc. of the 47th Design Automation Conference (DAC)*, pages 737–742. ACM, <http://chess.eecs.berkeley.edu/pubs/804.html>.
- [113] Liu, L., Liu, Y., Wang, X., Yu, D., Liu, K., Huang, H., and Hu, G. (2015). Developing an effective 2-d urban flood inundation model for city emergency management based on cellular automata. *Natural Hazards and Earth System Sciences*, 15(3):381–391, DOI: 10.5194/nhess-15-381-2015, <https://www.nat-hazards-earth-syst-sci.net/15/381/2015/>.
- [114] Lynch, N. A. (1996). *Distributed Algorithms*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN: 1558603484.
- [115] Mahamadou Traore, Ousmane Thiare, B. (2017). Insect dynamic cellular modeling and simulation. In *2017 Winter Simulation Conference (WSC)*.
- [116] Maraseni, T. N., Mushtaq, S., and Maroulis, J. (2009). Greenhouse gas emissions from rice farming inputs: a cross-country assessment. *The Journal of Agricultural Science*, 147(2):117–126, DOI: 10.1017/S0021859608008411.
- [117] May, D. (1983). Occam. *SIGPLAN Not.*, 18(4):69–79, ISSN: 0362-1340, DOI: 10.1145/948176.948183, <http://doi.acm.org/10.1145/948176.948183>.
- [118] Mele, K. (2013). Insect Soup Challenge: Segmentation, Counting, and Simple Classification. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 168–171. DOI: 10.1109/ICCVW.2013.28.
- [119] Millie Bond (2008). Grasshopper. <https://a-z-animals.com/animals/grasshopper/>.
- [120] Ministry of Natural Resources and Environment of Vietnam (2016). *Scenarios for climate change and sea level rise in Vietnam*. Vietnam environmental resources and maps. in Vietnamese.
- [121] Ministry of Natural Resources and Environment (Vietnam) (2016). Circulars of Hydro Meteorological observation content. http://chuyentrang.monre.gov.vn/upload/13377/fck/files/05-BTNMT_signed.pdf. in Vietnamese.

- [122] Miyamoto, K. (1964). Fish eye lens. *J. Opt. Soc. Am.*, 54(8):1060–1061, DOI: 10.1364/JOSA.54.001060, <http://www.osapublishing.org/abstract.cfm?URI=josa-54-8-1060>.
- [123] Nguyen, D., Clauss, K., Cao, S., Naeimi, V., Kuenzer, C., and Wagner, W. (2015). Mapping Rice Seasonality in the Mekong Delta with multi-year envisat ASAR WSM Data. 7:15868–15893.
- [124] Nguyen, V. and James, H. (2013). Measuring household resilience to floods: A case study in the Vietnamese Mekong River Delta. *Ecology and Society*, 18:13, DOI: 10.5751/ES-05427-180313.
- [125] Nguyen, V. G. N., Huynh, H. X., Vo, T. T., and Drogoul, A. (2011). On weather affecting to brown plant hopper invasion using an agent-based model. In *Proceedings of the International Conference on Management of Emergent Digital EcoSystems, MEDES '11*, pages 150–157, New York, NY, USA. ACM, ISBN: 978-1-4503-1047-5, DOI: 10.1145/2077489.2077517, <http://doi.acm.org/10.1145/2077489.2077517>.
- [126] Niazi, M. and Hussain, A. (2011). Agent-based computing from multi-agent systems to agent-based models: A visual survey. *Scientometrics*, 89(2):479–499, ISSN: 0138-9130, DOI: 10.1007/s11192-011-0468-9, <http://dx.doi.org/10.1007/s11192-011-0468-9>.
- [127] NISO (2004). Understanding metadata. Online, <http://www.niso.org/standards/resources/UnderstandingMetadata.pdf>. ISBN 1-880124-62-9.
- [128] NVIDIA (2014). Jetson tk1. http://elinux.org/Jetson_TK1.
- [129] Organization, W. M. (1985). A guide to the code form FM 92-IX Ext. GRIB. <https://www.wmo.int/pages/prog/www/WDM/Guides/Guide-binary-2.html>.
- [130] Otuka, A. (2013). Migration of rice planthoppers and their vectored re-emerging and novel rice viruses in east asia. *Frontiers in Microbiology*, 4:e5, DOI: doi:10.3389/fmicb.2013.00309, <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3836001/>.
- [131] Otuka, A., Watanabe, T., Suzuki, Y., Matsumura, M., Furuno, A., Chino, M., Kondo, T., and Kamimuro, T. (2006). A migration analysis of *Sogatella*

- furcifera (Horváth) (Homoptera: Delphacidae) using hourly catches and a three-dimensional simulation model. *Agricultural and Forest Entomology*, 8(1):35–47, ISSN: 1461-9563, DOI: 10.1111/j.1461-9555.2006.00284.x, <http://dx.doi.org/10.1111/j.1461-9555.2006.00284.x>.
- [132] Øye, O. K., Wein, W., Ulvang, D. M., Matre, K., and Viola, I. (2012). *Real Time Image-Based Tracking of 4D Ultrasound Data*, pages 447–454. Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-33415-3, DOI: 10.1007/978-3-642-33415-3_55, http://dx.doi.org/10.1007/978-3-642-33415-3_55.
- [133] Packard, N. H. and Wolfram, S. (1985). Two-dimensional cellular automata. *Journal of Statistical Physics*, 38(5):901–946, ISSN: 1572-9613, DOI: 10.1007/BF01010423, <http://dx.doi.org/10.1007/BF01010423>.
- [134] Pauwels, K., Rubio, L., and Ros, E. (2015). Real-time pose detection and tracking of hundreds of objects. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99):1–1, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2015.2430652.
- [135] Pearson, E. S. and Hartley, H. O., eds. (1972). *Biometrika Tables for Statisticians*. Cambridge University Press. ISBN 0-521-06937-8.
- [136] Pham, C., Lecuire, V., and Moureaux, J. M. (2013). Performances of multi-hops image transmissions on IEEE 802.15.4 wireless sensor networks for surveillance applications. In *2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 477–484. ISSN: 2160-4886, DOI: 10.1109/WiMOB.2013.6673402.
- [137] Phan, C. H., Huynh, H. X., and Drogoul, A. (2010). An agent-based approach to the simulation of brown plant hopper (BPH) invasions in the Mekong delta. In *2010 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), Hanoi, Vietnam, 1-4 November, 2010*, pages 1–6. IEEE, DOI: 10.1109/RIVF.2010.5633134, <http://dx.doi.org/10.1109/RIVF.2010.5633134>.
- [138] Phan Van Tan and Ngo Duc Thanh (2013). Climate change in Vietnam: Some research results, challenges and opportunities in international integration. *Earth and Environmental Sciences, Vietnam National University, Hanoi*, 29(2):42–55. in Vietnamese.

- [139] Potamitis, I. and Rigakis, I. (2015). Novel noise-robust optoacoustic sensors to identify insects through wingbeats. *IEEE Sensors Journal*, 15(8):4621–4631, ISSN: 1530-437X, DOI: 10.1109/JSEN.2015.2424924.
- [140] Pottier, M. (1995). Operational databases at meteo-france. In *Fifth Workshop on Meteorological Operational Systems, 13-17 November 1995*, pages 220–225, Shinfield Park, Reading. ECMWF, ECMWF.
- [141] Prasad, R. and Noubir, G. (1993). Implementing synchronous network model on a fip network. In *Advances in Parallel and Distributed Systems, 1993., Proceedings of the IEEE Workshop on*, pages 20–27. DOI: 10.1109/APADS.1993.588733.
- [142] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1988). *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, ISBN: 0-521-35465-X.
- [143] Pria, M. J., McElderry, H., Oh, S., Siddall, A., and Wehrell, R. (2008). Use of a video electronic monitoring system to estimate catch on groundfish fixed gear vessels in California: a pilot study. Archipelago Marine Research Ltd, British Columbia, Canada.
- [144] Puntanen, S. (2007). Linear Models and Generalizations: Least Squares and Alternatives, 3rd Edition by C. Radhakrishna Rao, Helge Toutenburg, Shalabh, Christian Heumann (with Contributions by Michael Schomaker). *International Statistical Review*, 75(3).
- [145] Qi, H., Jiang, C., Zhang, Y., Yang, X., and Cheng, D. (2014). Radar observations of the seasonal migration of brown planthopper (*Nilaparvata lugens* Stål) in Southern China. *Bulletin of Entomological Research*, 104(6):731–741, DOI: 10.1017/S0007485314000558.
- [146] Qing Yao, Jun LV, Qing-jie Liu, Guang-qiang Diao, Bao-jun Yang, Hong-ming Chen, and Jian Tang (2012). An Insect Imaging System to Automate Rice Light-Trap Pest Identification. *Journal of Integrative Agriculture*, 11(6):978 – 985, ISSN: 2095-3119, DOI: [http://dx.doi.org/10.1016/S2095-3119\(12\)60089-6](http://dx.doi.org/10.1016/S2095-3119(12)60089-6), <http://www.sciencedirect.com/science/article/pii/S2095311912600896>.
- [147] Rachmatullah, R. S. and Azizah, F. N. (2017). A conceptual data model for flood based on cellular automata using moving object data model. *Journal of Physics:*

- Conference Series*, 801(1):012037, <http://stacks.iop.org/1742-6596/801/i=1/a=012037>.
- [148] Raspberry PI (2016). Raspberry pi 3.0 benchmarks. <https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/>.
- [149] Rasshofer, R. H., Spies, M., and Spies, H. (2011). Influences of weather phenomena on automotive laser radar systems. *Advances in Radio Science*, 9:49–60, DOI: 10.5194/ars-9-49-2011, <https://www.adv-radio-sci.net/9/49/2011/>.
- [150] Reissig, W., Heinrichs, E., Litsinger, J., Moody, K., Fiedler, L., Mew, W., and Barrion, A. (1986). *Illustrated guide to integrated pest management in rice in tropical Asia*. IRRI, Philippines.
- [151] Research Program on Rice, Global Rice Science Partnership (GRiSP) (2013). Ricepedia, the online authority on rice. <http://ricepedia.org/rice-as-a-plant/growth-phases>.
- [152] Reyes, T. M. and Gabriel, B. P. (1974). The life-history and consumption habits of *Cyrtorhinus lividipennis reuter* (hemiptera: Miridae). *Philippine Entomologist*, 3(2):79–88.
- [153] Riley, J. R., Xia-Nian, C., Xiao-Xi, Z., Reynolds, D. R., Guo-Min, X., Smith, A. D., Ji-Yi, C., Ai-Dong, B., and Bao-Ping, Z. (1991). The long-distance migration of *Nilaparvata lugens* (Stål) (Delphacidae) in China: radar observations of mass return flight in the autumn. *Ecological Entomology*, 16(4):471–489, ISSN: 1365-2311, DOI: 10.1111/j.1365-2311.1991.tb00240.x, <http://dx.doi.org/10.1111/j.1365-2311.1991.tb00240.x>.
- [154] Robert Forchheimer, Per Ingelhart, C. J. (1992). MAPP2200: a second-generation smart optical sensor. DOI: 10.1117/12.58391, <http://dx.doi.org/10.1117/12.58391>.
- [155] Russo, L., Russo, P., and Siettos, C. I. (2016). A complex network theory approach for the spatial distribution of fire breaks in heterogeneous forest landscapes for the control of wildland fires. *PLOS ONE*, 11(10):1–18, DOI: 10.1371/journal.pone.0163226, <https://doi.org/10.1371/journal.pone.0163226>.

- [156] Ryu, Y., Verfaillie, J., Macfarlane, C., Kobayashi, H., Sonnentag, O., Vargas, R., Ma, S., and Baldocchi, D. D. (2012). Continuous observation of tree leaf area index at ecosystem scale using upward-pointing digital cameras. *Remote Sensing of Environment*, 126:116 – 125, ISSN: 0034-4257, DOI: <http://dx.doi.org/10.1016/j.rse.2012.08.027>, <http://www.sciencedirect.com/science/article/pii/S003442571200346X>.
- [157] Salcedo, E., Zheng, L., Phistry, M., Bagg, E. E., and Britt, S. G. (2003). Molecular basis for ultraviolet vision in invertebrates. *Journal of Neuroscience*, 23(34):10873–10878, ISSN: 0270-6474, <http://www.jneurosci.org/content/23/34/10873>.
- [158] Samet, H. and Tamminen, M. (1988). Efficient component labeling of images of arbitrary dimension represented by linear bintrees. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(4):579–586, ISSN: 0162-8828, DOI: 10.1109/34.3918, <http://dx.doi.org/10.1109/34.3918>.
- [159] Schwalbe, E. (2005). Geometric modelling and calibration of fisheye lens camera systems. In *Proceedings 2nd Panoramic Photogrammetry Workshop, Int. Archives of Photogrammetry and Remote Sensing*, pages 5–8.
- [160] Semtech (2017). LORA technology. <http://www.semtech.com/wireless-rf/rf-transceivers/>.
- [161] Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611, ISSN: 00063444, <http://www.jstor.org/stable/2333709>.
- [162] Sinnott, R. W. (1984). Virtues of the Haversine. *Sky and Telescope*, 68:158.
- [163] Smajgl, A., Toan, T., Nhan, D., Ward, J., Trung, N., Tri, L., Tri, V., and Vu, P. (2015). Responding to rising sea levels in the Mekong Delta. *Nature Climate Change*, 5(2):167.
- [164] Smirnov, N. (1948). Table for estimating the goodness of fit of empirical distributions. *Ann. Math. Statist.*, 19(2):279–281, DOI: 10.1214/aoms/1177730256, <http://dx.doi.org/10.1214/aoms/1177730256>.
- [165] Spalding, A. and Parsons, M. (2004). Light trap transects – a field method for ascertaining the habitat preferences of night-flying lepidoptera, using *mythimna turca*

- (linnaeus 1761) (lepidoptera: Noctuidae) as an example. *Journal Of Insect Conservation*, 8(2):185–190, ISSN: 1572-9753, DOI: 10.1007/s10841-004-1350-3, <http://dx.doi.org/10.1007/s10841-004-1350-3>.
- [166] Srivastava, A. N., Chawla, N. V., Yu, P. S., and Melby, P., editors (2010). *Proceedings of the 2010 Conference on Intelligent Data Understanding, CIDU 2010, October 5-6, 2010, Mountain View, California, USA*. NASA Ames Research Center, <http://c3.ndc.nasa.gov/dashlink/resources/220/>.
- [167] Stava, O. and Benes, B. (2010). Connected component labeling in cuda. *Hwu., WW (Ed.), GPU Computing Gems*.
- [168] Steinbach, M., Tan, P.-N., Kumar, V., Klooster, S., and Potter, C. (2003). Discovery of climate indices using clustering. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '03*, pages 446–455, New York, NY, USA. ACM, ISBN: 1-58113-737-0, DOI: 10.1145/956750.956801, <http://doi.acm.org/10.1145/956750.956801>.
- [169] Steinhäuser, K., Chawla, N. V., and Ganguly, A. R. (2010). Complex Networks In Climate Science: Progress, Opportunities And Challenges. In [166], pages 16–26, <http://c3.ndc.nasa.gov/dashlink/resources/224/>.
- [170] Steinhäuser, Karsten, Chawla, Nitesh V., and Ganguly, Auroop R. (2011). Complex networks as a unified framework for descriptive analysis and predictive modeling in climate science. *Statistical Analysis and Data Mining*, 4(5):497–511, ISSN: 1932-1872, DOI: 10.1002/sam.10100, <http://dx.doi.org/10.1002/sam.10100>.
- [171] Straw, A. D., Branson, K., Neumann, T. R., and Dickinson, M. H. (2010). Multi-camera real-time three-dimensional tracking of multiple flying animals. *Journal of The Royal Society Interface*, ISSN: 1742-5689, DOI: 10.1098/rsif.2010.0230, <http://rsif.royalsocietypublishing.org/content/early/2010/07/13/rsif.2010.0230>.
- [172] Taillandier, P., Vo, D.-A., Amouroux, E., and Drogoul, A. (2012). *GAMA: A Simulation Platform That Integrates Geographical Information Data, Agent-Based Modeling and Multi-scale Control*, pages 242–258. Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-25920-3, DOI: 10.1007/978-3-642-25920-3_17, http://dx.doi.org/10.1007/978-3-642-25920-3_17.

- [173] Takagi, H., Thao, N. D., and Anh, L. T. (2016). Sea-Level Rise and Land Subsidence: Impacts on Flood Projections for the Mekong Delta's Largest City. *Sustainability*, 8(9), ISSN: 2071-1050, DOI: 10.3390/su8090959, <http://www.mdpi.com/2071-1050/8/9/959>.
- [174] Takagi, M. and Tajima, J., editors (1992). *Proceedings of IAPR Workshop on Machine Vision Applications, MVA 1992, December 7-9, 1992, Tokyo, Japan*.
- [175] Tarjan, R. E. (1975). Efficiency of a good but not linear set union algorithm. *J. ACM*, 22(2):215-225, ISSN: 0004-5411, DOI: 10.1145/321879.321884, <http://doi.acm.org/10.1145/321879.321884>.
- [176] Teodora Sanislav, L. M. (2012). Cyber-physical systems - concept. *Challenges and Research Areas. CEAI, Vol.14, No.2,*, pages 28-33.
- [177] Thang, H. Q., Phuong, L. A., Raedt, L. D., Deville, Y., Bui, M., Linh, T. T. D., Oanh, N. T., Sang, D. V., and Ngoc, N. B., editors (2015). *Proceedings of the Sixth International Symposium on Information and Communication Technology, Hue City, Vietnam, December 3-4, 2015*. ACM, ISBN: 978-1-4503-3843-1, <http://dl.acm.org/citation.cfm?id=2833258>.
- [178] The United Nations Economic and Social Commission for Asia and the Pacific (ESCAP) (2016). Disasters in asia and the pacific: 2015 year in review. <http://www.unescap.org/resources/disasters-asia-and-pacific-2015-year-review>.
- [179] Tian, D. and Georganas, N. D. (2002). A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, WSNA '02*, pages 32-41, New York, NY, USA. ACM, ISBN: 1-58113-589-0, DOI: 10.1145/570738.570744, <http://doi.acm.org/10.1145/570738.570744>.
- [180] Tomassini, M., Giacobini, M., and Darabos, C. (2005). Evolution and dynamics of small-world cellular automata. *Complex Systems*, 15.
- [181] Topaz, C. M., Bernoff, A. J., Logan, S., and Toolson, W. (2008). A model for rolling swarms of locusts. *The European Physical Journal Special Topics*, 157(1):93-109, ISSN: 1951-6401, DOI: 10.1140/epjst/e2008-00633-y, <https://doi.org/10.1140/epjst/e2008-00633-y>.

- [182] Tran, H. V., Huynh, H. X., Phan, V. C., and Pottier, B. (2016). A federation of simulations based on cellular automata in cyber-physical systems. *EAI Endorsed Trans. Context-aware Syst. & Appl.*, 3(7):e3, DOI: 10.4108/eai.12-2-2016.151086, <https://doi.org/10.4108/eai.12-2-2016.151086>.
- [183] Truong, T. M., Amblard, F., Gaudou, B., Sibertin-Blanc, C., Truong, V. X., Drogoul, A., Huynh, H. X., and Le, M. N. (2013). An implementation of framework of business intelligence for agent-based simulation. In *Proceedings of the Fourth Symposium on Information and Communication Technology*, SoICT '13, pages 35–44, New York, NY, USA. ACM, ISBN: 978-1-4503-2454-0, DOI: 10.1145/2542050.2542069, <http://doi.acm.org/10.1145/2542050.2542069>.
- [184] Truong, V. X. (2014). *Optimization by Simulation of an Environmental Surveillance Network : application to the Fight against Rice Pests in the Mekong Delta (Vietnam)*. (*Optimisation par simulation de réseaux de surveillance environnementale : application à la lutte contre les insectes ravageurs du riz dans le delta du Mekong (Vietnam)*). PhD thesis, Pierre and Marie Curie University, Paris, France, <https://tel.archives-ouvertes.fr/tel-01337323>.
- [185] Truong, V. X., Huynh, H. X., Le, M. N., and Drogoul, A. (2012). Modeling a surveillance network based on unit disk graph technique - application for monitoring the invasion of insects in mekong delta region. In Rahwan, I., Wobcke, W., Sen, S., and Sugawara, T., editors, *PRIMA 2012: Principles and Practice of Multi-Agent Systems - 15th International Conference, Kuching, Sarawak, Malaysia, September 3-7, 2012. Proceedings*, volume 7455 of *Lecture Notes in Computer Science*, pages 228–242. Springer, DOI: 10.1007/978-3-642-32729-2_16, http://dx.doi.org/10.1007/978-3-642-32729-2_16.
- [186] Truong Chi Quang, Vo Quang Minh, Nguyen Thi Hong Chau, and Ho Van Chien (2013). Managing the brown planthopper caught by light traps for supporting of rice cultivation in the mekong delta. In *National GIS application conference*. (in Vietnamese).
- [187] Tsai, Y. R. (2008). Sensing coverage for randomly distributed wireless sensor networks in shadowed environments. *IEEE Transactions on Vehicular Technology*, 57(1):556–564, ISSN: 0018-9545, DOI: 10.1109/TVT.2007.905624.

- [188] Tsonis, A. and Roebber, P. (2004). The architecture of the climate network. *Physica A: Statistical Mechanics and its Applications*, 333:497 – 504, ISSN: 0378-4371, DOI: <http://dx.doi.org/10.1016/j.physa.2003.10.045>, <http://www.sciencedirect.com/science/article/pii/S0378437103009646>.
- [189] United Nations (1992). United Nations Framework Convention on Climate Change. <https://unfccc.int/resource/docs/convkp/conveng.pdf>.
- [190] United Nations (1998). Kyoto Protocol to the United Nations framework convention on climate change. <http://unfccc.int/resource/docs/convkp/kpeng.pdf>.
- [191] United Nations (2015). Paris Agreement. http://unfccc.int/files/essential_background/convention/application/pdf/english_paris_agreement.pdf.
- [192] United Nations Country Team in Viet Nam (2016). Vietnam Consolidated Report on Drought and Saltwater Intrusion (Reporting period: Oct 2015 - Mar 2016). <https://reliefweb.int/sites/reliefweb.int/files/resources/Vietnam>
- [193] University of Lincoln (2015). Monitoring nesting seabirds using computer vision. <http://gtr.rcuk.ac.uk/projects?ref=EP/H017143/1>.
- [194] Vaishampayan, S. M., Kogan, M., Waldbauer, G. P., and Woolley, J. T. (1975). Spectral specific responses in the visual behavior of the greenhouse whitefly, *Trialeurodes vaporariorum* (Homoptera: Aleyrodidae). *Entomologia Experimentalis et Applicata*, 18(3):344–356, ISSN: 1570-7458, DOI: 10.1111/j.1570-7458.1975.tb00407.x, <http://dx.doi.org/10.1111/j.1570-7458.1975.tb00407.x>.
- [195] Vaithilingam, D. C. (2013). *Role of insect light trap in organic agriculture*. Romvijay Biootech Private Limited.
- [196] Vincent, M. K. (1984). Conceptual aspects of environmental monitoring. <http://www.iwr.usace.army.mil/Portals/70/docs/iwrreports/STAFFPAPER84-SP-3AUGUST1984.pdf>. U.S. Army Engineer Institute for Water Resources, Water Resources Support Center, Casey Building Fort Belvoir, Virginia 22060.
- [197] Vinh, P. C. and Barolli, L., editors (2016). *Nature of Computation and Communication - Second International Conference, ICTCC 2016, Rach Gia, Vietnam*,

- March 17-18, 2016, Revised Selected Papers*, volume 168 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer, ISBN: 978-3-319-46908-9, DOI: 10.1007/978-3-319-46909-6, <http://dx.doi.org/10.1007/978-3-319-46909-6>.
- [198] Waidyanatha, N. (2010). Towards a typology of integrated functional early warning systems. *IJCIS*, 6(1):31–51, DOI: 10.1504/IJCIS.2010.029575, <https://doi.org/10.1504/IJCIS.2010.029575>.
- [199] Wallace, F., Williams, K., Towler, R., and McGauley, K. (2015). *Innovative Camera Applications for Electronic Monitoring*. Alaska Sea Grant Bookstore, <http://doi.org/10.4027/fbgics.2015.06>.
- [200] Warren, B. (2001). *Photography*. Student Material TV Series. Delmar Publishers, ISBN: 9780766817777, <https://books.google.fr/books?id=sbdGeFem1zwC>.
- [201] Wassmann, R., Hien, N. X., Hoanh, C. T., and Tuong, T. P. (2004). Sea level rise affecting the vietnamese mekong delta: Water elevation in the flood season and implications for rice production. *Climatic Change*, 66(1):89–107, ISSN: 1573-1480, DOI: 10.1023/B:CLIM.0000043144.69736.b7, <https://doi.org/10.1023/B:CLIM.0000043144.69736.b7>.
- [202] Woiwod, I., Gould, P., and Conrad, K. (2011). The Rothamsted light-trap network. <https://www.rutlandwater.org.uk/the-rothamsted-light-trap-network/>.
- [203] Woiwod, I. P. and Harrington, R. (1994). Flying in the face of change: The Rothamsted Insect Survey. In Leigh, R. A. and Johnston, A. E., editors, *Long-term Experiments in Agricultural and Ecological Sciences*, pages 321–342. CAB International. ISBN 0-85198-933-0.
- [204] Wolfram, S. (1983a). Statistical mechanics of cellular automata. *Reviews of Modern Physics*, 55(3):601–644, DOI: 10.1103/revmodphys.55.601, <http://dx.doi.org/10.1103/revmodphys.55.601>.
- [205] Wolfram, S. (1983b). Statistical mechanics of cellular automata. *Reviews of Modern Physics*, 55:601–644, DOI: 10.1103/RevModPhys.55.601, <http://link.aps.org/doi/10.1103/RevModPhys.55.601>.

- [206] Wolfram, S. (1984). Cellular automata as models of complexity. *Nature*, 5985(5):419–424, DOI: 10.1038/311419a0, <http://dx.doi.org/10.1038/311419a0>.
- [207] Wolfram, S. (July 1982). Cellular automata as simple self-organizing systems. *Caltech preprint CALT*, pages 68–938.
- [208] Worf, D. L. (1980). *Biological monitoring for environmental effects / edited by Douglas L. Worf*. Lexington Books Lexington, Mass, ISBN: 0669033065.
- [209] World Meteorological Organization, 2nd ed. Geneva (1992). International Meteorological Vocabulary (WMO-No. 182). Also available online through METEOTERM http://www.wmo.int/pages/prog/lsp/meteoterm_wmo_en.html.
- [210] Yan, X. and Su, X. G. (2009). *Linear Regression Analysis: Theory and Computing*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, ISBN: 9789812834102, 9812834109.
- [211] Yingshu Li, My T. Thai, and Weili Wu (2008). *Wireless Sensor Networks and Applications*. Springer.
- [212] Yoshihashi, R., Kawakami, R., Iida, M., and Naemura, T. (2015). Construction of a bird image dataset for ecological investigations. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 4248–4252. DOI: 10.1109/ICIP.2015.7351607.
- [213] Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, ISSN: 0162-8828, DOI: 10.1109/34.888718, <http://dx.doi.org/10.1109/34.888718>.

Résumé

Réseaux de capteurs sans fil pour l'observation du climat et de la biologie dans une région tropicale d'agriculture intensive Méthodes, outils et applications pour le cas du Delta du Mékong, Vietnam

Les changements climatiques ont des impacts considérables sur le temps, les océans et les rivages, la vie sauvage. Ils amènent des problèmes désormais considérés comme majeurs par les gouvernements et organisations internationales.

Ces efforts ont fourni un cadre à cette thèse, qui propose de procéder en boucle fermée de l'observation d'insectes ravageurs, avec des centaines de capteurs en réseau ("light traps"), au système d'information, et enfin à des décisions de lutte, manuelles ou automatiques. Le point d'appui pratique est la conception d'un système de comptage d'insectes proliférant dans les cultures de riz (BPH). L'abstraction que nous développons est celle d'une machine environnementale de grande taille, distribuée, qui capte et synthétise l'information, élabore des connaissances, et prend des décisions. Autour de cette abstraction, nous avons élaboré un système de vision "fisheye" effectuant le comptage des insectes.

Nous proposons un système d'information géographique directement connecté au réseau de capteurs.

Le couplage direct, "cyber-physique", entre les systèmes d'information et l'observation de l'environnement à échelle régionale est une nouveauté transposable, qui permet de comprendre et contrôler quantité d'évolutions.

Mot-clés: *changement climatique, système de vision, piège lumineux, caméra fisheye, classification comptage d'insectes, Cicadelle brune, machine environnementale, automates cellulaires, réseau synchrone, système d'information géographique*

Abstract

Sensors and wireless networks for monitoring climate and biology in a tropical region of intensive agriculture Methods, tools and applications to the case of the Mekong Delta of Vietnam

Climate changes bring problems related to nature evolutions. Global warming has an impact on sea level, weather patterns, and wild life. A number of national and international organizations are developing research programs in these directions, including threats on cultures and insect proliferation. Monitoring these phenomena, observing consequences, elaborating counteracted strategies are critical for the economy and society.

The initial motivation of this work was the understanding of change impacts in the Mekong Delta region. From there, automatic observation tools were designed with a real time information system able to integrate environmental measures, then to support knowledge production.

Tracking environment evolutions is distributed sensing, which can be the association of efficient sensors and radio communications, operated under the control of an information system. Sensing insects is very complex due to their diversity and dispersion. However, this is feasible in the case of intensive agricultural production as it is the case of rice, having a small number of pests. An automatic vision observatory is proposed to observe the main threats for the rice, as an evolution of manual *light traps*. Radio communication weaves these observatories into a network with connection to databases storing measures and possible counteractions. An example observatory has a fisheye camera and insect counting algorithms for the BPH practical case in Vietnam.

By considering the observation system as an input for an abstract machine, and considering decision and actions taken as a possible control on the environment, we obtain a framework for knowledge elaboration that can be useful in lots of other situations.

Keywords: climate change, vision system, light trap, fisheye lens vision, insect counting, Brown Planthopper, environment machine, cellular automata, synchronous network, geographic information system

Ces travaux sont partie intégrante du projet STIC-Asie SAMES financé par le Ministère des Affaires Etrangères (UBO/LabSTICC et al.). Le gouvernement du Vietnam a financé la thèse en liaison avec l'Université de Can Tho (CTU).