

CRANFIELD UNIVERSITY

Faisal Khan

**Swarm Eye: A Distributed Autonomous Surveillance System**

School of Applied Sciences

M.Sc. by Research  
Academic Year: 2010- 2011

Supervisor: Jorn Mehnen

CRANFIELD UNIVERSITY

School of Applied Science

M.Sc. by Research  
Academic Year 2010 - 2011

Faisal Khan

**Swarm Eye: A Distributed Autonomous Surveillance System**

Supervisor: Jorn Mehnen  
2011

© Cranfield University 2011. All rights reserved. No part of this publication may be reproduced without the written permission of the copyright owner.

## **Abstract**

---

Having precise information in complex, dynamic, highly demanding and stressful situations is of utmost importance for quick, objective and well informed decision making. Knowing the exact position of a unit in complex, unknown or confusing environment such as ragged mountain areas or narrow labyrinthine alleys of a city can be very challenging and is of high relevance in tactical decision making. Conventional means such as GPS and satellite photos are important information sources but provide only limited and static information. In tactical situations rich 3D images and dynamically self-adapting information are needed to overcome this restriction; this information should be collected where it is available. Distributed ground teams or swarms of UAVs can provide different and dynamic views of a tactical scene. Swarms are sets of interconnected units that can be arranged and coordinated in any flexible way to execute a specific task in a distributed manner.

The Swarm Eye is a concept that provides a platform for combining the powerful techniques of swarm intelligence, emergent behaviour and computer graphics in one system. Swarm Eye allows the testing of new image processing concepts for a better and well informed decision making process. By using advanced collaboratively acting eye units, the system can observe, gather and process images in parallel to provide high value information. To capture visual data from an autonomous airborne unit, the unit has to be in the right position in order to get the best visual sight. This Swarm Eye system also provides autonomous formations for UAV or airborne units to form a better formation autonomously in distributed manner in accordance with the situation.

## Acknowledgements

---

This thesis is based on the research work done for a project sponsored by Lockheed Martin. I would like to thank to Dr. Jorn Mehnen for giving me the opportunity to work on my master thesis at Cranfield University. It is an honor to work under his supervision. I really appreciate Dr. Jorn Mehnen's advise and supervision. I am also grateful to Dr. Toby Breckon for being the subject advisor and providing his expertise advise on Computer Graphics.

Special thanks to Dr. Paul Townsend (Industry side project manager) for providing the sponsorship and direction of the project.

My utmost thanks to my family for supporting me during my studies, in particular to my parents without them, I would not have been able to pursue my masters.

This research is funded by an industrial sponsor and the findings of this project are confidential, therefore this report will not contain all the findings/algorithms and sensitive materials.



## Statement of originality

---

The work presented in this thesis is the result of original research done by myself, in collaboration with my supervisor, subject advisor and industrial sponsor while enrolled as a Master by research student at Cranfield University in the Department of Applied Sciences. The project has utilised various open source technologies stated in the reference list

This report does not state all the findings/contributions of this research project “Swarm Eye” as per the confidential agreement with the defence related industrial sponsor.

Student Name: \_\_\_\_\_ Faisal Khan \_\_\_\_\_

*Faisal Khan*

Signature: \_\_\_\_\_

# Document Control

---

## Version History

Status / Version	Date Issued	Amendments
4.0	05/05/2011	Final Thesis

# Contents

---

1	Introduction .....	1
1.1	The Swarm Eye.....	1
1.2	The Comparability of Insect Compound Eye and Swarm Eye .....	2
1.3	Background of the Project.....	3
1.4	Aims .....	3
1.5	Objectives .....	3
1.6	Benefits of the Swarm Approach .....	4
1.7	Out of scope .....	4
1.8	Potential Applications of the Swarm Eye Project.....	5
1.9	Summary.....	5
2	Terminology and Definitions .....	6
2.1	Swarm Intelligence.....	6
2.2	Swarm Eye .....	6
2.3	Formation .....	6
2.4	Surveillance System .....	6
2.5	Autonomous Systems.....	6
2.6	Coordination.....	6
2.7	Swarm Units .....	6
3	Related Work .....	7
3.1	Introduction.....	7
3.2	Present Research.....	7
3.3	Swarm organisation and Formation.....	7
3.4	Multi Robot coordination.....	8
3.4.1	Leader Follower Approach.....	9
3.4.2	Behaviour Based Approach.....	9
3.4.3	Virtual Structure Approach.....	10
3.5	Composite Eyes and their Computer Simulation .....	10
3.6	Sensor Network .....	11
3.7	Particle Swarm and Optimisation.....	12
3.8	Detection of road vehicles from an unmanned aerial vehicle.....	12
3.9	Image Processing .....	13

3.10	Motion Detection.....	15
3.11	Determination of Relative Position .....	15
3.12	Research Aim .....	16
3.13	Research Questions .....	17
3.14	Summary.....	17
4	Methodology.....	18
4.1	Introduction.....	18
4.2	Motivation .....	18
4.3	Approach / Methodology .....	18
4.4	Swarm Communication Approach.....	18
4.4.1	Centralised .....	19
4.4.2	Decentralised Communication .....	19
4.4.3	The Communication Protocol .....	20
4.5	Swarm Eye Formation.....	20
4.6	Visibility Factor Index .....	21
4.7	Formation Algorithm .....	22
4.8	Initial Formation .....	22
4.9	Dynamic Formation.....	23
4.10	Design of Formation Algorithm .....	23
4.10.1	Dynamic formation making .....	24
4.11	Summary.....	25
5	Swarm Eye Prototype Software.....	27
5.1	Introduction.....	27
5.2	Swarm Eye Software Design.....	28
5.3	Visualisation Unit Design.....	30
5.4	Camera/Eye Unit Software Design .....	31
5.5	The communication structure .....	32
5.6	Virtual environment.....	34
5.7	Image Rendering Package.....	35
5.8	Collective Image Representation .....	36
5.8.1	Panoramic image generation.....	36
5.8.2	Swarm Formation and Panoramic Images .....	38
5.9	Determine Relative Positions .....	43
5.9.1	Determination of the observer position within a scene.....	43
5.9.2	Determination of a target position within a scene.....	45
5.9.3	Moving Target Detection.....	48
5.9.4	Swarm Formation .....	48
5.9.5	Transformation Steps .....	51
5.10	UAV Physical Manoeuvring Constraints .....	53

5.11	Present Work .....	54
5.12	Implementation Performance Analysis .....	54
5.13	Future Work.....	55
6	Conclusion.....	57
7	References .....	59
8	Glossary of Terms and Acronyms .....	63

# List of Figures

---

Figure 1: The basic structure of compound eyes. [46] .....	2
Figure 2: Six perspective images (a) and cube projection (b) (from [10]).....	11
Figure 3: <i>Detection of road vehicles from UAV [15]</i> .....	13
Figure 4: Focus stacking (left: with focus stacking, right: without focus stacking) .....	13
Figure 5: Research areas covered in the Swarm Eye Project. ....	16
Figure 6 : Centralised System.....	19
Figure 7: Decentralised System.....	19
Figure 8: Architecture of a Decentralised Swarm Eye Flying Formation Algorithm .....	21
Figure 9: This figure of virtual scene demonstrates the visibility factor index execution .....	22
Figure 10: the demonstration of full swarm dividing into small sub groups. 16 unit swarm been divided into 4 small groups of 4 units. ....	23
Figure 11: Decentralised Dynamic Algorithm block diagram. As each unit will hold this algorithm.....	24
Figure 12: Swarm Eye Software Design .....	29
Figure 13: Visualisation Unit Graphical User Interface. (Shows a list of four registered eye units. By ticking the checkboxes you can activate or deactivate individual eyes through the Visualisation Unit.) .....	30
Figure 14: Swarm Eye Distributed Communication .....	33
Figure 15: The scene generated by using md2 object. ....	34
Figure 16: Views as seen by four independently running Swarm Eye units. ....	35
Figure 17: Panoramic view generation in Swarm Eye. ....	36
Figure 18: Views of 4 different eye units covering a tactical scene.....	37
Figure 19: Panoramic image created by combining the images from Figure 17. ....	37
Figure 20: Panoramic View Yielded by swarm in curve formation .....	38
Figure 21: The orientation of swarm units in curve formation.....	39
Figure 22: Wider area coverage by swarm in arrow formation .....	39
Figure 23: Swarm in arrow formation with downward camera orientation .....	40
Figure 24: The circle formation of panoramic image with viewing outside the circle camera orientation ....	40
Figure 25: Swarm in circle formation focusing towards outer circle.....	41
Figure 26: Swarm in semi circle formation focusing towards inner circle .....	41

Figure 27: Swarm Circular formation with all swarms focusing towards the centre of circle ..... 42

Figure 28: A distorted panorama generated using SIFT and RANSAC ..... 42

Figure 29:: Triangulation (resection) using a single eye to determine the position  $(x_1, y_1)$  of the eye itself given two positions  $(x_2, y_2)$  and  $(x_3, y_3)$ , respectively, and the angles  $\alpha$  and  $\beta$  relative to a north-south orientation on a map. .... 44

Figure 30: Determining the individual position by triangulation. North-south directions are indicated by the blue-white pointers. The angles  $\alpha$  and  $\beta$  as well as the positions of the houses are known..... 44

Figure 31: Determining an unknown position of a third object by triangulation. North-south directions are indicated by the blue-white pointers. The angles  $\alpha$  and  $\beta$  as well as the positions of the eyes 1 and 2 are known. .... 45

Figure 32: Determining the position of an unknown object by triangulation. The position of the jet (Swarm Eye at the wheel of the jet), a Swarm Eye unit (small crosshair in the middle of the screen) floating in the sky as well as their relative view angles to the target object are known. The target in the image is a small dynamically moving four-by-four car hidden behind the palm tree (one wheel is visible). The Swarm Eyes can track the car while it is moving, update its relative position and send the data to the Visualisation Unit which displays the view as seen here. .... 46

Figure 33: Global positioning using information about the Field of View (FoV). Left: calculation of the dimensions of an object by using at the relative proportion of the dimensions of the object within the FoV. Right: coordinate system with relative positions of points at the grid crossings. .... 47

Figure 34: Determination of the position and dimension of objects in Swarm Eye by proportionate dimension calculations..... 47

Figure 35: The user interface allows for selection of different formations and swarm units..... 48

Figure 36: Swarm units in the arrow formation..... 49

Figure 37: The Line formation of the swarm units, where each aircraft demonstrates a swarm unit. .... 50

Figure 38 : The Arrow formation of the swarm units. .... 50

Figure 39: Dynamic Formation Figure and their explanation ..... 51

Figure 40 Block Diagram of Dynamic Formation Algorithm ..... 52

Figure 41: Dynamic Formation Sequence Diagram of basic objects..... 53

---

# **1 Introduction**

---

This chapter will give an introduction on Swarm Eye project. This chapter also highlights the main aims, objectives, Swarm Eye concept and background of the project.

## **1.1 The Swarm Eye**

The Swarm Eye philosophy follows the idea that a set of individuals can form a very powerful system when joined and self-organised in a swarm. The individuals in a swarm can communicate and interact with each other directly or indirectly. The Swarm Eye project provides a software platform for testing this swarm philosophy. The Swarm Eye system consists of several independently running Swarm Eye units that can communicate in any specified way. The units can also take photos while navigating through a simulated environment. The images can be combined in any fashion for generating high quality panoramic images, for 3D image generation, for automatic position determination in case of GPS malfunctions in a tactical situation or for movement detection in a scene. The intelligent combination of multiple units gathering images and processing them at the same time leads to techniques that can make use of the synchronicity of parallel images processing such as speed determination, 3D image generation of rapidly moving objects, movement detection of partially hidden objects and much more.

Swarm Eye follows the idea of using nature to shape swarms. Swarms are structures caused by the emergent behaviour of individuals. Natural swarms prove to be very effective while being robust against variations in the environment or even loss of individuals from the swarm. Swarms follow rules that can lead to extremely complex behaviour. The rules are often surprisingly simple and the components of the swarm are very basic. However, the emergent behaviour leads to very powerful and effective results. The Swarm Eye system is a distributed system therefore the loss of a few individuals may compromise the performance, but the system will be in working order.

The Swarm Eye is a tactical decision support system that mainly uses optical information. It can be airborne or land based. The swarm consists of a scalable number of rather simple units that collaborate in a self-organised way. The Swarm Eye concept is designed for medium to large scale swarms. The swarm unit numbers can range from a couple to several hundred units. The Swarm Eye is designed to operate in the field, where several sources of information are needed to make well informed tactical decisions.



## 1.2 The Comparability of Insect Compound Eye and Swarm Eye

A compound eye can consist of hundreds of individual photoreceptors. The image perceived is a combination of inputs from the numerous ommatidia (individual "eye units"), which are located on a convex surface, thus pointing in slightly different directions. Each ommatidia consists of a lens, transparent crystalline cone, light sensitive visual cells and pigment cells.

To compare with simple eyes, compound eyes possess a very large view angle and can detect fast movement and, in some cases, the polarization of light. Because the individual lenses are so small, the effects of diffraction impose a limit on the possible resolution that can be obtained. This can be countered by increasing lens size and number. To see with a resolution comparable to our simple eyes, humans would require compound eyes which would each reach the size of their head [18].

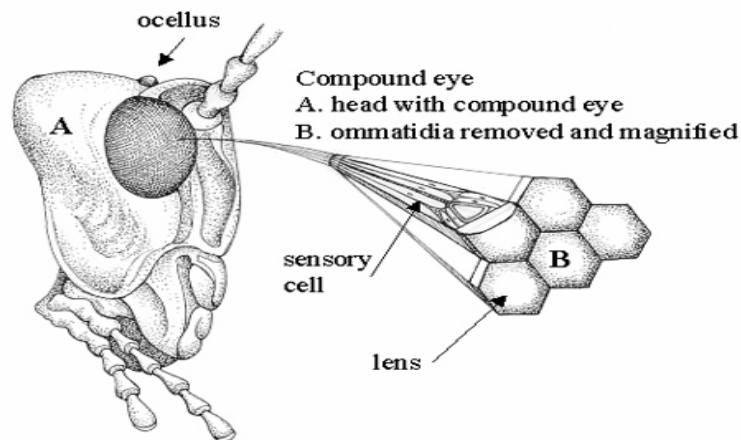


Figure 1: The basic structure of compound eyes. [46]

The compound eye is excellent at detecting motion. As an object moves across the visual field, ommatidia are progressively turned on and off. Because of the resulting "**flicker effect**", insects respond far better to moving objects than stationary ones. Honeybees, for example, will visit wind-blown flowers more readily than still ones.

Hence, the concepts adapted from Compound Eye to Swarm Eye are:

- To have many eyes /camera
- Observe the object from many different angles

In Swarm Eye the camera locations are in different places, whilst in compound eye each ommatidia is next to each other.

These are the comparable units between compound eyes and Swarm Eye project.

### 1.3 **Background of the Project**

Nature has been a source of inspiration for today's various modern and complex technology and intriguing systems. Many inventions have been done by adopting the natural system into either mechanical or electric manner. Swarm Eye is also a nature inspired project. The research project "Swarm Eye: A Distributed Autonomous Positioning and Target Detection System" has been adopted from nature. Swarms are schools of fish, colonies of ants and termites, flocks of birds, herds of land animals. Insects such as mosquitoes, spiders and dragon flies have a compound eye which provides them with a large view angle. This helps them to quickly detect fast movements and compensate for the physical lacks of the single units by "intelligent" combination of information.

### 1.4 **Aims**

The aim of the research is to investigate the potential of the swarm concept in tactical decision making by delivering a prototype system. This includes:

- Analysis of the concept of natural swarm's behaviour into Swarm Eye system.
- Development of a computer simulation to demonstrate the potentials of image processing and swarm intelligence /artificial intelligence techniques in the Swarm Eye system.
- Design an algorithm that controls the formation of the swarm to facilitate in eliminating the blind spots of the tactical scene.

The prototype Swarm Eye system is to act as a platform for exploration of further ideas in relation to swarm implications within tactical decision making.

### 1.5 **Objectives**

The objective of the proposed research is to explore how swarm like objects can aid today's tactical decision making. The specific objectives of this project are:

- Design and implement computer simulation to model the swarm like objects
- Self-organisational swarm objects which provide the visual information of the required terrain.
- Analyse natural swarm features such as swarm formation and swarm collaboration for Swarm Eye.
- Present an effective communication protocol for Swarm Eye.
- Present autonomous positioning and targeting detection system prototype which provides multi-angle 3D visual information of the targeted terrain.

- Implement image stitching and motion detection features into the computer simulation.

## 1.6 **Benefits of the Swarm Approach**

The main properties of any natural/artificial swarm are expected to be the following

- Scalable(range could be few hundreds) numbers of simple individual units
- Generally robust and more reliable
- Task Sharing
- Distributed and decentralised system which provides robust and organised visual data.
- Autonomous system requires less human effort.
- Multiple eyes provide multiple independent views of 3D visual data.

There are many more benefits of the swarm approach depending on the place the swarm is being used. The above are the more general benefits.

## 1.7 **Out of scope**

This project merges diverse techniques into one platform in order to increase the performance, robustness and usability. Some parts of image processing have been investigated in this project but are limited to the context of panoramic image generation, image analysis and pixellation. The elements that are outside the scope in this project are the following items.

- Particle Swarm Optimisation
- Implementing or analysing a Hardware Architecture of Swarm Unit
- Swarm Optimisation Algorithms
- Robotics
- Machine learning
- Biological aspects of Swarm behaviour
- Swarm Intelligence

The Swarm Eye system utilises the existing techniques of image processing and Swarm intelligence but does not contribute to these fields.

## 1.8 **Potential Applications of the Swarm Eye Project**

This Swarm Eye Project can facilitate many different fields such as swarms of UAVs or underwater robots for military, to manage environmental and industrial disasters, environmental monitoring, search and rescue actions or surveillance or land survey and many more.

## 1.9 **Summary**

The Swarm Eye takes concepts from nature to adapt dynamically to varying environments. The Swarm Eye system mainly uses optical information. It can be airborne or land based. The swarm consists of a scalable number of rather simple units that collaborate in a self-organised way. The Swarm Eye is designed for medium to large scale swarms.

The goal of this project is to design a system which provides more robust, effective, detailed information for tactical decision making. The project achieves this by using advanced technologies in the fields of images processing and swarm intelligence.

This system could be used for military, police, surveillance, space and other industries.

This project will focus on the software side of the formation algorithm, image processing (image analysis, pixilation), and swarm intelligence. Furthermore this project will focus on the effects of autonomous swarm formation of UAVs.

## **2 Terminology and Definitions**

---

This chapter will contain major definitions and the terms that have been used in this thesis and related research. This chapter will define these terms in the related context.

### **2.1 Swarm Intelligence**

The collective and emergent intelligence of groups of autonomous agents without having a central controller. Implicit in such a definition is the fact that there is assumed that each unit perform simple tasks to provide a solution to a complicated problem. Different research defined the swarm intelligence in different ways but the above definition is the mean core of all.

### **2.2 Swarm Eye**

Swarm Eye is the name of this research project and Eye is a camera/swarm Unit which could be airborne or land based. An eye will mainly captures visual data by using cameras.

### **2.3 Formation**

In this project, formation mainly indicates to flying formation of the swarm units. The flying formation is when two or more flying objects make any order or shape by collaborating motion with each other.

### **2.4 Surveillance System**

Surveillance systems are those systems which could monitor a dynamic situation in context of time, which is mainly used by military. Surveillance systems are also some times called reconnaissance system. The project Swarm Eye could also be used as a Surveillance or Reconnaissance system. The Swarm Eye research project will aid present surveillance systems.

### **2.5 Autonomous Systems**

Autonomous Systems refers to those robots or machines which can operate and perform a given task without human input. In this project the tasks that are performed autonomously are forming the formation, motion detection and the generation of panoramic images.

### **2.6 Coordination**

The coordination in this project refers mainly to swarm units working together for a goal or effect. Coordination may also be described as communication between the swarm units to form a formation.

### **2.7 Swarm Units**

In this thesis Swarm Units refer to UAVs (Unmanned Aerial Vehicle, MAVs (Micro Aerial Vehicles), ground cameras and ground vehicles with the capabilities of capturing the visual data, and data communication and intelligence. Swarm Units could also be robots.

## **3 Related Work**

---

### **3.1 Introduction**

This chapter explores related work within the sub components of a functional swarm system. It attempts to highlight these components and their working to achieve the application of swarm behaviour in robotics and aerial UAVs. Most of the work discussed here arises from the current work in the field of swarm robotics and is capable of being applied to the swarm of aerial UAVs. The work presented here distinguishes swarm sub-components into the categories of formation and coordination of a swarm. The application of this coordinated activity in tactical decision making also requires the application of suitable image processing techniques. The chapter also covers these image processing techniques and its background.

### **3.2 Present Research**

The Swarm Eye system utilises many different technologies such as image analysis, swarm intelligence, multi-robot coordination, and autonomous systems. Therefore the research has to be carried out in these fields in order to use the appropriate technology for the Swarm Eye. This chapter will show a brief research on image processing, Intelligence and autonomous systems. The appropriate detail research will be shown in swarm formation, multi-robots coordination and swarm self-organisation.

### **3.3 Swarm organisation and Formation**

A major problem with a centralised system is that if the centre of the system becomes dysfunctional then the whole system stops working. Due to this, the swarm must be decentralised and autonomous. In Swarm Eye if one or more individual members are not functioning even then the system should operate on those members which satisfy minimum working requirement. To make a swarm autonomous, a more robust and adaptable system from computation intelligence algorithms could be applied [20].

In swarms that occur naturally, the formation of pulling chains by weaver ants or the surrounding of a prey like a target by a group of predators can be seen as examples of multi pattern formation.

An implementation of such an approach can be seen in the swarm intelligence based multi-robot pattern formation [28]. This model argues that the pattern to be formed is controlled by the surroundings. At the same time, the kind of shape and the size of the final patterns formed are partially decided by the task or objective of each agent during the coordination.

A different method [28] relies on the robots themselves to make decisions instead of a centralised coordinator. The communication and coordination among the robots is realised through a Virtual Pheromone Model and a modified Particle Swarm Optimisation (PSO) algorithm [28].

Without centralised coordination, the multi-robot system works in a complete distribution mode, using the local environmental information and virtual pheromone to allocate tasks among the robots. In this model, each robot has two working modes during the task: exploration and dispersion. Initially, the grid-based map is divided into sub-areas with some grids noted as the predefined pattern. In the exploration working mode, the robot computes its optimised target sub-area which contains noted grids, basing on the virtual pheromone given by other robots, to find the nearest sub-area in which the least robots already move, and make the movement basing on PSO model. Upon reaching the targeted sub-area, it turns into the dispersion working mode to place itself on a certain noted grid. With this swarm intelligence based approach, the multi-robot system can form the predefined pattern in an effective and scalable way [28].

In the approach used by The Burchelli *et al.* [21] the focus is on collective intelligence. The rules used to organise their swarm raids has broad application for advancement in programming of multi-agent autonomous systems, the design and for providing insight into the concepts of collective self-organisation and intelligence. [21]

The insect colonies distribute resources and tasks to each unit in order to collectively solve difficult problems instead of using any centralised control. The behaviour comes from the local interactions with no global knowledge and it tends to be a simple set of rules.

These behavioural algorithms are usually robust to alter, keeping their effectiveness in extremely diverse environment or recourse distribution over a broad range of tasks [21]

There are three key features in the army ant foraging system that are vital to well designed multi-agent systems as well:

Robustness: army ants can tolerate major colony disturbance without significant impact to colony behaviour;

Scalability: system behaviour is maintained across vast variation in the system size;

Flexibility: army ants can dynamically allocate individuals and tasks as needed and adapt to different environments [21].

The usage of GPS in the formations of autonomous airships has been inspected by Olsen *et al.* [39]. Experiments were carried out using two blimps, both equipped with multi carrier-phase GPS (CPGPS) receivers allowing the establishment of attitude and location of the craft. The experiment identified formations with attitude errors. The errors were in the range of less than five degrees with further positional errors of less than a foot.

### 3.4 **Multi Robot coordination**

The approaches to multi-robot coordination are mainly the following:

- The Leader-follower approach

- The Behaviour based models
- The Virtual structure techniques

### **3.4.1 Leader Follower Approach**

The leader follower approach has been widely used in robotics. In this model one or more robots act as the leader and other units follow the leader's position while maintaining a predetermined distance from the leaders [31]

### **3.4.2 Behaviour Based Approach**

The behaviour-based approach to multi-robot coordination integrates several task-oriented behaviours that act at the same time toward their respective tasks.

There is much different behaviour possible, for example the avoidance of obstacles, or maintaining a position within the formation, tracking the nearest neighbour etc. Reynolds [32] behaviour-based simulations of flocks of birds were one of the first in its kind. His "Boids" model showed how simple behaviours that effectively use only local sensing could generate "emergent" group behaviour.

In [31] "emergent" group behaviour was investigated, where behaviours like aggregation, dispersion and avoidance could be used to create a "flocking" global behaviour in mobile robots.

In [34] some of the initial tests were prepared on formation following behaviours on robots that were mobile. Metrics were used to determine how well the robots maintained the formations. Some techniques for measuring each robot's relative position as well as four different formations were studied.

A nature inspired [35] approach allows control of multiple units by associating them to a particular attachment site. This control arrangement is similar to the formation of crystals by the arrangement of molecules within nature. The simulation of this arrangement using groups of eight units and three formations showed that efficient formation keeping is subject to local sensing.

Work discussed in [36] highlights robot formation based on neighbouring interaction and sensing. The scheme is similar to the leader follower approach but allows each robot to maintain a fixed orientation and distance from a sensed robot which is not required to be a leader.



### 3.4.3 Virtual Structure Approach

This approach treats the complete formation as a compact structure. This structure is manipulated by using a three step algorithm that defines the exploitation of the formation.

Step 1: The desired global coordinates of the formation are established.

Step 2: Individual units within the structure are assigned their own transformations.

Step 3: The control laws for each unit are derived. These include the basic geometric arrangements [37], formation feedback control [38] and the virtual structure architecture [30] that has demonstrable features.

### 3.5 Composite Eyes and their Computer Simulation

How compound eyes in animals work has been described extensively in literature. Miller [5] shows that the compound eye is excellent in detecting motion. Different facets of the eye are progressively stimulated when objects move across the visual field of the insect. This is what we call the “flicker effect”, which makes the insect respond far better to moving objects than to unmoving ones. This notion has been adapted to Swarm Eye to detect motion more sensitively.

Struwe [6] shows that compound eyes like that of a moth can also show a very high sensitivity to light. This makes some insects, like the praying mantis, a good hunter even during the early and late hours of the day.

Bishop [7] shows that some insects such as the drone fly can see ultraviolet as well as natural colour light. This helps drone flies to forage together with the honeybees they are mimicking.

The compound eyes of most insects have many hundreds of lens facets. For example the fruit fly *Drosophila melanogaster* has about 700 facets. Each facet samples only one small portion in the insect's visual field. This provides these insects with a very wide field of view. Buschbeck *et al.* [8] show that composite lens eyes of some insects (e.g. *Xenos peckii*) having no more than 50 facets can still show excellent performance. The issue of seeing the world multiple times with different view angles is compensated by the cortex of these insects.

Koshy *et al.* [9] Describes a design of a composite eye for computer vision for detecting moving objects in a closed environment. They designed a small unmoving array of seven off-the-shelf cameras and show that it is able to track a moving object with this kind of system. The 3D position of the object can also be determined in the overlap areas.

Neumann [10] simulates the properties of a compound eye. His model includes a spherical field of view, overlapping Gaussian-shaped receptive fields, a singular viewpoint, and a space-variant receptor distribution. The algorithm creates low resolution spherical images

from multiple static perspectives. For representing spherical images, Neumann projects the 3D images onto e.g. cubes (see Fig. 2) because current raster graphics technology is optimised to construct planar, perspective images [10].

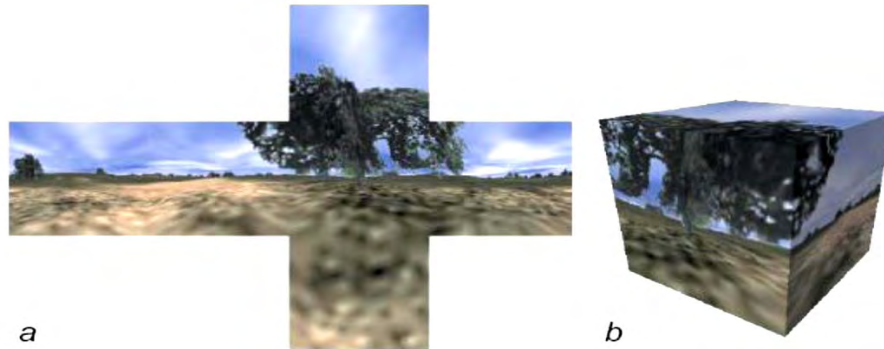


Figure 2: Six perspective images (a) and cube projection (b) (from [10])

### 3.6 Sensor Network

The wireless sensor network and multi mobile robots are both collective systems, with distributed and autonomous individuals. Sensor Network or Wireless Sensor Network algorithms are closely related to the Swarm Eye project in terms of autonomous behaviour and swarm formation.

Wenfeng [47] discuss the collective behaviour of wireless Sensor network (WSN) and multi mobile robots (MMR) and swarm features. The work discuss the challenges and the combination of the sensor networks and mobile multi robots and presents the layered dual-swarm framework which is a 3D communication structure that could be inherited in traditional swarm technology as well as building an efficient communication channel for swarms to cooperatively work together.

Wenfeng reports three kinds of combinations of Systems:

- 1- WSN and MMR. In this system, mobile robots can be used to spread and maintain the nodes of WSN [47]. With the support of WSN, robots can more effectively navigate in an unknown environment. And PEG (Pursuit Evasion Game) problem could be solved with global search [47].
- 2- Mobile Sensor Networks. The nodes have mobility as robots and the network is self-organised as WSN. The system can work more actively than WSN with static nodes [47].
- 3- Wireless sensor and actuator networks (WSAN). In this kind of networks, sensor nodes and actuator nodes communicate at the same network level.

Wenfeng provides strategy of combining the sensor network and mobile robots and their challenges, communication scheme, and control strategy of the system.

Murphy *et al.* [48] addresses the difficulties and risks of manual operation of wireless airborne sensor networks in unknown urban environments, besides describing the roles in a rescue team in detail. Approaches to decrease the number of roles required for operating multiple UAVs are introduced and options for equipping the UAVs with the ability to accomplish certain tasks autonomously are discussed.

Formation control is one approach to comprehend cooperative coordination by Ryan *et al.* [49]. Multi-UAV formation flight combines the research of both UAV and coordination, so it has increase concentration from both unmanned system and control communities.

Cooperative coordination is defined as requiring that a group of unmanned aerial vehicles to follow a predefined trajectory for flight missions while using their on-board sensors to acquire useful information while maintaining a specified formation pattern.

The flight path can be a set of waypoints or a predefined fly zone with boundaries. Because formation flights of a UAV fleet can significantly increase the global security and universal efficiency of the entire system, it can benefit most of the applications which are handled by a single UAV.

### **3.7 Particle Swarm and Optimisation**

The optimal positioning of a camera in 3D space using Particle Swarm Optimisation (PSO) has been addressed by Burelli *et al.* in 2008 [11]. They propose a system that adjusts the position of a *single* camera in a virtual environment automatically so that the view complies optimally with user defined constraints. Burelli uses a PSO algorithm to solve externally – i.e. from outside the scene – the associated camera positioning problem.

There exists several nature motivated optimisation techniques. PSO is a very well known and powerful algorithm for single-objective continuous parameter optimisation. NSGA-II and SPEA2 [12] are the state-of-the-art multi-objective optimisation algorithms (MOEA), i.e. algorithms where several objectives can be solved at the same time. The probably most advanced multi-objective optimiser may be MOEA/D [13]. This algorithm is a combination (hybrid) of several techniques. The reference [14] covers crucial aspects regarding the implementation of MOEA in industrial applications

### **3.8 Detection of road vehicles from an unmanned aerial vehicle**

This Swarm Eye project is based on the visual data which is mainly captured by camera (eye) and those cameras can also be mounted on the UAVs or any other flying object. In that aspect the issues in mounting cameras (eye) on the flying object should be considered.

The article “Autonomous Real-time Vehicle Detection from a Medium-Level UAV” written by Breckon [15] highlights some of the issues in capturing data from flying objects. This article highlights the vehicles detection from UAVs / aerial view camera which would be the feature of the Swarm Eye research project

The aerial detection of people benefits from the consistent projection of the upright human form to a vertical trace within a perspective view image. The general problem of aerial vehicle detection is also made significantly more challenging by the non-uniformity of vehicle colour, localised shape characteristics and overall dimension [15].

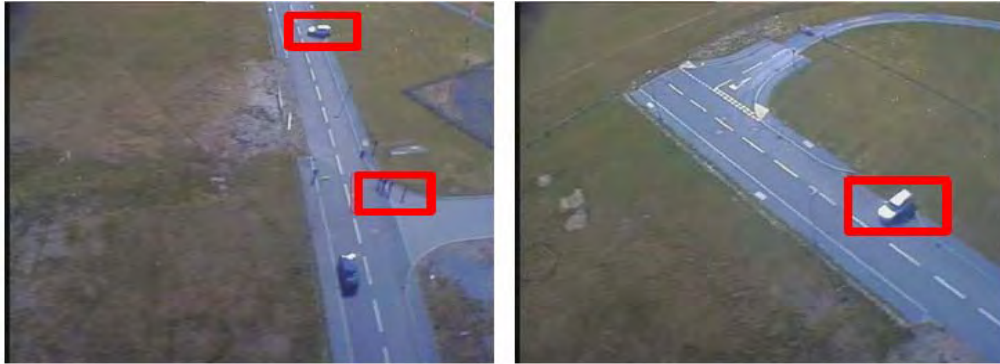


Figure 3: Detection of road vehicles from UAV [15]

A two staged approach to autonomous vehicle detection has been proposed by [15]: 1) primary detection using multiple trained cascaded Haar classifiers and 2) secondary verification based on UAV altitude driven vehicle size constraints [15].

### 3.9 Image Processing

A very useful technique for combining many static pictures into one larger image is called “image stitching”. With this technique several pictures of a scene having slightly different view angles are put together into one single image. This way one can generate e.g. panoramic views of landscapes. The difficulty here is to find the right camera positions so that the pictures overlap sufficiently well to allow the automatic stitching algorithm to operate properly [3]. Another image editing technique called “focus stacking” generates images that do not suffer from blur effects due to out of focus issues. Here several pictures are taken of the same object with different focus settings and the whole picture is composed using the in-focus parts of the picture only. An example of focus stacking is shown in Figure 4. The image on the left in Figure 4 is in focus at all points of the picture while parts of the image on the right are partially out of focus.



Figure 4: Focus stacking (left: with focus stacking, right: without focus stacking)

Focus stacking is just one of several methods that go under the more general term of HDR (High Dynamic Range) [4]. Scenes can have an extensive dynamic band of light to dark tones that cannot be apprehend by today's cameras using a single image. HDR is used to generate images that try to recreate what the eye sees by computing a photorealistic image by taking the information of multiple photos into account.

The collective image representation task is there to increase the visibility of the scene and reduce the blind spots for the user. The automatic stitching of two or more images requires a slight overlap between the images.

In order to perform efficient image stitching, the right camera positions [5] must be established. Furthermore a suitable image stitching algorithm must be applied to acquire the stitched panorama.

There are several ways for stitching images together to get one big image i.e. taking the image of one object to build a 3D object, taking images of big scenes into several small images in order to get a high resolution image, or taking images by rotating the camera itself to capture the surrounding of the camera.

The notion of combining many static image sources has been taken on by radio telescope arrays which use techniques such as aperture synthesis to vastly increase resolution. By interfering with the signals, these arrays generate images that have the quality of a virtual telescope of the size of the whole array. The largest array is the Giant Metre-wave Radio Telescope (GMRT) consisting of 30 antennas [30]. The European large telescope array LOFAR (LOW Frequency ARray) is currently being built using 15,000 small antennas. Software emulates a telescope that covers an area of about 230 miles in diameter [31].

M. Brown *et al* [29] have proposed Image stitching using unmoving features. They put forward a new system for automatic panorama stitching by the use of constant local features and a probabilistic model to verify image matches. This allows for recognition of multiple panoramas from unordered image set. This process is fully automatic without requiring any user input. The system is robust to camera zoom, changes in illumination due to flash and exposure/aperture settings and orientation of the input images. A multi-band blending scheme ensures smooth transitions between images despite lighting differences, whilst preserving high frequency details [29].

All the discussed approaches however suffer from the issues of:

- Scene motion
- Radial distortion
- Camera motion, small motions of the optical centre [29].

The Swarm Eye is an autonomous system therefore it would be convenient to use invariant feature based image stitching. This approach uses SIFT (Scale-Invariant Feature Transform) and RANSAC (Random Sample Consensus) algorithm to stitch the images.

### 3.10 Motion Detection

The Swarm Eye system is mainly based on visual cameras and exploits the analysis on the images fed by the cameras in real-time. The Swarm Eye system offers a very sensitive and effective motion detection system because Swarm Eye has multiple eyes/cameras on key locations. More cameras in the right locations mean fewer blind spots. Most of the motion detection algorithms are based on comparing the frames with each other and some also compare the frames with each other with the dimensions of time to extract the moving object's speed [19].

After motion detection the most important part is to identify the moving object. The identification of the object autonomously is a complex task as most real environments have many variations which can affect the image and visibility of the scene. The work of Breckon *et al.* [15] proposed the method of autonomous real-time vehicle detection from a medium-level. The detection of moving vehicles is vital in tactical decision making for military.

The general problem of aerial vehicle detection is also made significantly more demanding due to the non-uniformity of vehicle colour, localised shape characteristics and overall dimension. [15].The approach is based on two stages for autonomous vehicle detection:-

- 1) Primary detection using multiple trained cascaded Haar classifiers [15].
- 2) Secondary verification based on UAV altitude driven vehicle size constraints [15].

### 3.11 Determination of Relative Position

To determine the relative position within an unknown terrain is a challenging task. The DGPS /AGPS receivers make a fine selection for UAV positioning main sensors, because they are able to reach accuracy up to few centimetres using carrier-phase measurements. As they are the only position sensors that are generally used for UAV positioning. The reliability of their measurements is critical for UAV missions. [27]

To increase the reliability of the system, Fault Detection and Identification (FDI) techniques contribute to essential part [27]. Within the context of UAVs positioning G. Heredia [27] discusses about the problem of D-GPS and proposes FDI. Ideally, FDI uses all the information available to detect malfunctions in UAV sub-systems. However not all positioning errors can be detected by using all the sensors onboard the UAV. In missions with multiple UAVs it is possible to use the capabilities of the team of UAVs to augment the FDI system of each individual UAV. This way, the other UAVs' sensors can be used to get additional information that can be used by the UAV FDI system to detect errors in its own sensors. [27].

This approach proposes a precise positioning system but still suffers from electronic jamming.

Considering the vulnerabilities of GPS and the autonomous nature of swarm, our approach is to identify the object’s location by calculating the distance from the camera to the object using a laser range finder. This can be further enhanced by acquiring these distances from multiple perspectives and then applying trigonometric operations to calculate precise angles to achieve accurate location measures.

### 3.12 Research Aim

The primary goal of this research is to investigate and apply the concept of natural swarms (i.e. swarm of honey bee, swarm of termite/ants/fishes etc) that can be applied within UAV based swarms to make tactical decision making easier.

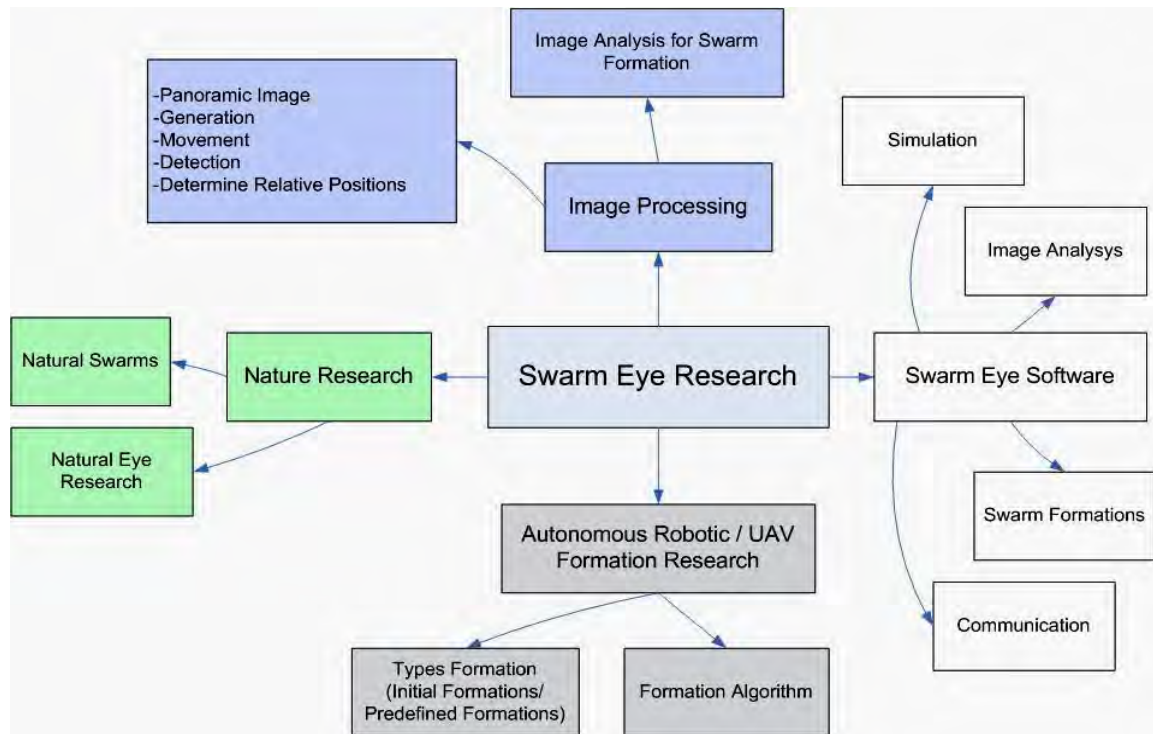


Figure 5: Research areas covered in the Swarm Eye Project.

Since UAVs are mainly used for aerial based surveillance the scope of our work is to identify suitable swarm techniques that can be applied within the context. This requires detailed research into the swarm formation, intelligence sharing and collaboration between individual members of the swarm itself. Since aerial surveillance data is produced in the form of an image, the work also looks into acquisition, interpretation and presentation of digital image using the techniques of digital image processing.

### 3.13 **Research Questions**

- How can various advanced technologies i.e. image processing, swarm intelligence and autonomous systems be incorporated together to create a suitable simulation platform for the analysis of the swarm application within tactical decision making?
- How can a swarm of UAVs efficiently be used to reduce the blind spot of an acquired scene?

### 3.14 **Summary**

The chapter summarised the work related to the fields of swarm formation and coordination. It also highlighted related work with regards to image processing techniques and its application in aiding tactical decision making.



## **4 Methodology**

---

### **4.1 Introduction**

The previous chapter discussed the current development in the related areas. This chapter demonstrates the progress of this research and the contribution of this project as well as identifying the approach that is taken in this research.

### **4.2 Motivation**

The precision of astronomic telescopes today have been improved multiple times by joining telescopes together in large worldwide networks. The analysis of visual data from distributed sources yields an unprecedented picture quality due to the synergetic effects of collaboration.

The compound eyes of insects are evolutionary proof of the efficiency of distributed visual systems. While each single eye is of limited power, the compound eye has a very large view angle. It can detect fast movements quickly and compensates for the physical lacks of the single units by intelligent combination of information.

The precise and quick orientation of a team working in an unknown, dynamic and hostile terrain as well as autonomous identification of moving targets is of utmost importance for tactical decision making. The Global Positioning System (GPS) is one of the most valuable and advanced but also vulnerable assets for guiding ground based or airborne vehicles. However, GPS provides only limited information which might prove insufficient in complex tactical situations.

### **4.3 Approach / Methodology**

Swarms are sets of small units which are interconnected and that can be arranged and coordinated in any flexible way to perform a specific task. A well coordinated swarm can efficiently process information from different distributed sources providing high quality data.

To test this “Swarm Eye” natural proven idea/concept in further detail, a prototype software tool is created with several cameras with interconnected parallel communication architecture; Image stitching and motion deduction with 3D coordinates. At this stage of the project 3D simulation will be used instead of real cameras due to resource limitations. The designed Swarm Formation algorithm will also be tested in this prototype software simulation.

### **4.4 Swarm Communication Approach**

Communication plays a vital role in the swarm. There are various strategies and approaches proposed for communications for swarm units that fall under the two main categories.

#### 4.4.1 Centralised

In the centralised communication approach, every unit in the swarm communicates with one centralised hub and the central hub sends commands/messages to all other units. The drawback of this approach is that if the centralised hub fails then the whole system will eventually fail.

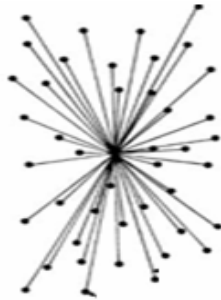


Figure 6 : Centralised System

However, this mechanism allows for fewer messages and in order to communicate with each other, the units' only need to communicate through the hub. This approach is also easier to manage due to the availability of the centralised hub.

#### 4.4.2 Decentralised Communication

There are different schemes to implement a decentralised communication model. In general, decentralised communication requires each unit to broadcast the message to every unit. Therefore the messages are interchanged more often as compared to a centralised system.



Figure 7: Decentralised System

These systems are more robust as they do not rely on any central hub of the system. They are also fault tolerant as the failure of a communicating unit does not adversely affect the functionality of the whole system.

In the Swarm Eye, the adopted communication approach is distributed decentralised. This is due to the requirements of autonomous operations and fault tolerance.

### 4.4.3 The Communication Protocol

The communication protocol is a very important part of this project. In order to have a decentralised communication all eyes are bound to have information shared between them. Most network based communication is either UDP or TCP based, a comparison of the two is provided below.

TCP	UDP
<ul style="list-style-type: none"><li>• Connection-Oriented</li><li>• Reliable- in delivery of messages</li><li>• Keep track of order (or sequence)</li><li>• Use checksums for detecting errors Remote procedures are not idempotent</li><li>• Congestion control mechanism is implemented by TCP</li></ul>	<ul style="list-style-type: none"><li>• Connectionless</li><li>• Unreliable- No attempt to fragment messages</li><li>• No reassembly, no synchronisation and no acknowledgment</li><li>• Remote procedures are idempotent</li><li>• UDP itself does not avoid congestion, and congestion control measures are implemented at the application level.</li></ul>

The UDP protocol does not support the guaranteed delivery of message. Therefore, the Swarm Eye uses TCP protocol as it allows guaranteed message delivery on the other end.

Furthermore, Swarm Eye user interface design register the eyes/cameras into to system and associate the eyes into group so when user would like to communicate, it can be done via communicating with certain eyes. The Swarm Eye uses the MPICH2 library for delivery of messages between the components of the system. MPICH2 also supports multiple communication protocols including TCP and shared memory.

### 4.5 Swarm Eye Formation

The identification and implementation of the proper formation of swarms is a vital element for efficient swarm application. These formations typically occur during the various phases of the swarm lifecycle.

The initial formation refers to the state of the swarm upon its creation; however this initial formation can be changed accordingly based on the swarm deriving parameters such as visibility. This adoption of a new formation overcomes any drawbacks associated with the static formation limitations. Within the context of aerial surveillance this dynamic formation is a desirable behaviour as it aids in overcoming obstacle avoidance and any static restrictions. It also allows for maximum coverage of any area of surveillance from the sky.

The formation is derived by a suitable algorithm that is applied on the swarm members to achieve a particular orientation of these members within the swarm. To aid this process this thesis proposes a *visibility factor index driven* algorithm to apply to a suitable swarm formation. This algorithm comprises of several steps as depicted in Figure 8.

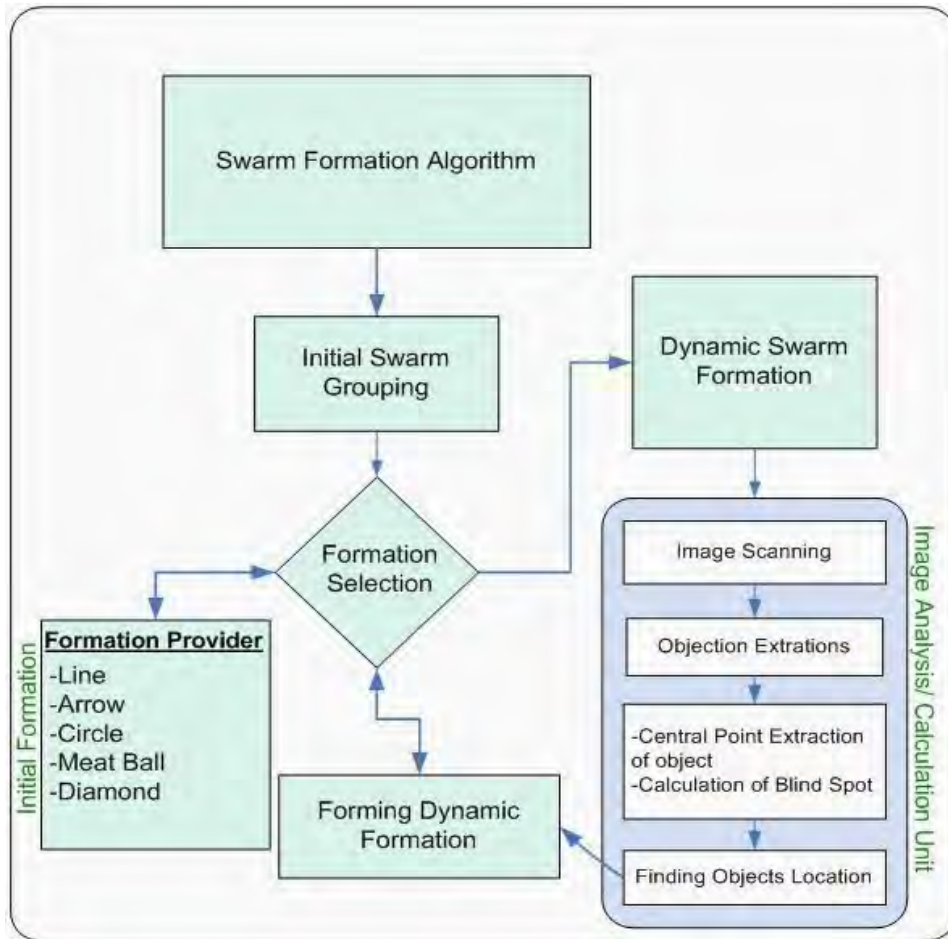


Figure 8: Architecture of a Decentralised Swarm Eye Flying Formation Algorithm

These steps aid in achieving a suitable formation for the swarm. The algorithm aims to apply a suitable formation to the swarm so that maximum visibility can be achieved over an area of surveillance. There are two main categories of these formations, *initial* and *dynamic*. The next section discusses these two formation categories in detail.

#### 4.6 Visibility Factor Index

The visibility factor will be decided depending on how much of the required object is visible to the individual eye. Each eye will have a rank. The eye that will cover more area of required object that eye will be ranked higher in the Visibility Factor Index (VFI). This task is explained more elaborately in the figure below.

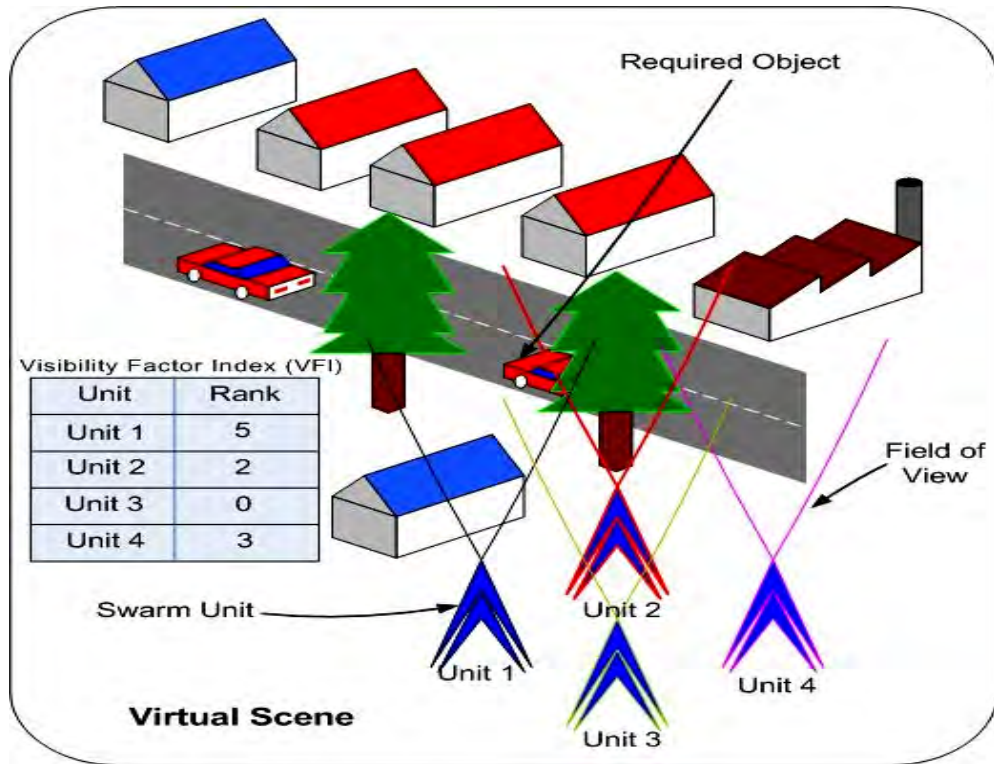


Figure 9: This figure of virtual scene demonstrates the visibility factor index execution

The above figure 9 shows four swarm units, each unit with its own point of view. The unit numbered one can see more of the required object (the car behind the tree), therefore in the ranking table it is ranked higher than others. The VFI ranking is dependent on the visibility, therefore, higher visibility yields higher VFI values and vice versa. The VFI aids the swarm system to establish swarm units with higher visibility at their present location. On the other hand, VFI allows for selection of low VFI swarm units for relocation in order to improve their VFI. The swarm formation algorithm also utilises low VFI swarm units as candidates for location transformation and swarm formation.

#### 4.7 Formation Algorithm

The formation can be implemented in different ways in this project. The swarm formation is divided into following two sub categories.

- a) Initial Formation
- b) Dynamic Formation.

#### 4.8 Initial Formation

In this Swarm Eye system, the initial formation refers to the formation which will be adopted by the swarm at the start up. As soon as the unit will be airborne and get registered into the swarm they will form a formation to fly, which could be arrow, diamond, semi-circle, circle formations etc.

This initial formation can be decided based on a few factors, for example if a swarm unit intends to save fuel then they will adopt the arrow formation [42, 43, and 44].

In the case of poor visibility, the formation will switch into a dynamic formation offering maximum visibility. The dynamic and initial formation algorithms are explained later in the report.

## 4.9 Dynamic Formation

The dynamic formation will be adopted by the swarm if the required objects are not visible due to blind spots in the current formation. This dynamic formation will be selected by the algorithm presented in the later section of the thesis.

## 4.10 Design of Formation Algorithm

The proposed formation algorithm presented in this section aims to derive a suitable formation based on the *visibility factor index*. This abstract term refers to a quantifiable unit that can measure visibility with regards to a particular *object of interest* within an area of surveillance. The objective of the algorithm is to apply a suitable formation to the swarm so that maximum collaborated visibility is achieved. The algorithm divides a swarm into several sub-swarms applying suitable formations at both the sub-swarm and the overall swarm level. This approach allows for individual sub-swarms within a main-swarm to have their own formation. With the main swarm divided into multiple individual sub-swarms each can have its own unique formation. This allows for a mix of formations within a main swarm that can be changed accordingly as desired.

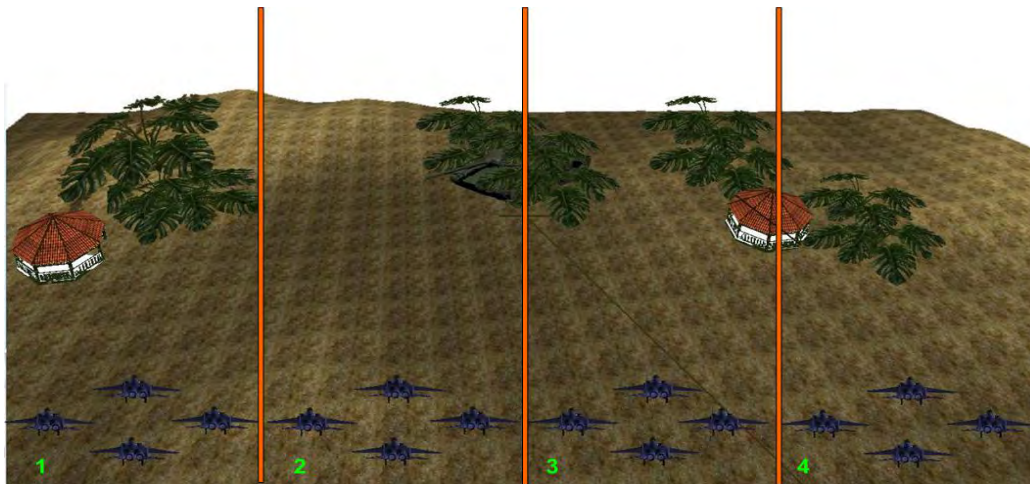


Figure 10: the demonstration of full swarm dividing into small sub groups. 16 unit swarm been divided into 4 small groups of 4 units.

The algorithms attempt to create sub-swarms of similar number of units by applying the  $\sqrt{U}$  function.  $U$  denotes the total number of the units in the swarm. For example, if the total numbers of units are 16 then applying  $\sqrt{16}$  will yield 4. The formation will make 4 groups, each group containing 4 units (in other words  $4 \times 4 = 16$  total swarm units) as shown in the

above figure 10. In the case of any left over unit, these units will join the nearest group. Each group can be used for different purposes as well as share the same task and each group can perform a certain job or task. This  $\sqrt{u}$  function approach has been selected due to the simplicity of implementation and it also allows for creation of equal number of groups with similar number of units. A fair distribution of tasks to individual units is also guaranteed by this approach. Various other approaches such as clustering can also be applied as a replacement to this approach. However the intent of this function is allocate units to swarm and if required it can be easily replaced by more efficient similar function.

#### 4.10.1 Dynamic formation making

Once all the units are assigned a group, each group will search its own allocated task. If the group faces an object/terrain with blind spots it will scan all the eyes visual index numbers. The eye which has the best ranked visual index of the required object will scan the required object's image and calculate the centre of the object (ObjC) from its own view. After locating the centre point of the object it will calculate the distance (dBS) between the ObjC and blind spot (BS) and after calculating this distance (dBS) the system will scan the visual index again. The eye which has the lowest visual index and is closest to the destination point will shift its position to a surveillance point which will allow it to cover the previously hidden area of the object. This process will iterate itself until the blind spot of the required object is optimally covered. Overview of the algorithm is also available in the figure 11 below.

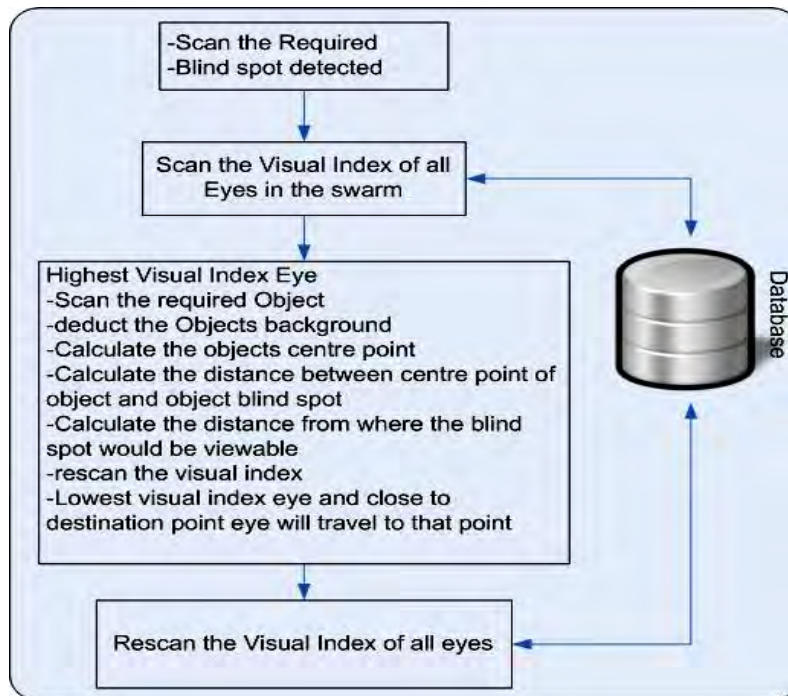


Figure 11: Decentralised Dynamic Algorithm block diagram. As each unit will hold this algorithm

This algorithm has been partially implemented and is not fully optimised at this stage. Specifically there may be a case of repeated iterations to cater for the dynamicity of the



scene. These iterations must be optimised to a certain scope but this issue has been left for future work. However, it allows for the exploration of this novel approach within tactical decision making scenarios.

In a previous chapter it has already been discussed that there are a few algorithms related to swarm formation and optimisation. The Autonomous Swarm Formation is a relatively new topic. As shown in the related work chapter various algorithms have been developed on swarm formation by different approaches [30, 31, 32, 33, 34, and 35].

There are many developments made in swarm intelligence, swarm formation, swarm optimisation. The detailed literature review (chapter 4) shows many developments in related fields with their own limitations. This Swarm system attempts to overcome some limitation in the following provided manner:

- The usage of all the technologies (such as use of image processing, swarm formation, relative position relevance etc) used on the Swarm platform
- The Unique Swarm Formation based on Visual Index to eliminate the blind spots or to cover hidden objects of the tactical scene.

This Swarm system also has a few limitations and drawbacks. There are many algorithms which have to be implemented, such as obstacle avoidance, keeping the formation, collision avoidance etc. Without these algorithms this system will not perform as designed. This swarm requires a high amount of data communication. These swarm formations intend to have units from 16 to 250, more or less units will decrease the performance of the whole swarm. Although the algorithm and the whole system has been designed to cater for a high amount of units, small changes could be adopted to make the performance optimal if more units are added to the swarm. The formation algorithm would not design the best optimal formation in every scenario, although the algorithm would provide better results in most cases. The full testing has not been done to the formation algorithm to provide results at this stage.

#### **4.11 Summary**

The chapter discussed the development of swarm formation algorithm which is the highlighted contribution of this project. The presented novel algorithm utilises advanced technologies such as image analysis, parallel communication, and computer intelligence to achieve efficient swarm formation. This formation can be initial or dynamic and is primarily driven by the visibility. The units in the swarm utilise a decentralised communication model and operate in an autonomous manner. Another novel aspect of the algorithm is to divide the swarm into groups of sub-swarms with an equal number of units. This allows for a two tier swarm formation modelling where a sub-swarm may have a different formation than another sub-swarm within the main swarm. This approach explores a mixed formation model that is applied at both the sub and the main swarm. This project contributes a new



idea of swarm formation based on the *visibility factor index* to cater for the need of an optimal visibility driven swarm. This approach can be used in swarm of UAVs to aid in intelligence gathering and tactical decision making.

## 5 Swarm Eye Prototype Software

### 5.1 Introduction

This chapter will demonstrate the software design, development and testing of the Prototype Software of Swarm Eye.

The Swarm Eye software prototype demonstrates the concept of the Swarm Eye of interacting autonomous devices that can organise themselves, take pictures of a tactical scene and detect moving objects in that scene. The main objective of the proposed software tool project is to supply the research sponsor (industrial sponsor) with a proof of concept of the Swarm Eye idea.

In order to manage the software side of the project the software development has been broken into several small unit packages that comprise as follows:-

- Designing
- Virtual Environments
- Eye Unit Software Structures
- Data Exchange Interfaces
- Image Rendering
- Collective Image Representation
- Determine Relative Positions
- Moving Targets Detection
- Self-organised Optimisation
- Tests, Prototype

#### **Designing**

Designing of the “Swarm Eye” software includes gathering full software requirements, analysis of technology available and formal or logic design with comparison of software requirements.

#### **Virtual environment**

A computer scenario of a tactical environment refers to the simulation which would simulate the camera with parallel communication with GUI.

**Eye unit software structure**

Client brief and protocols Implementation are the basis of the software structure. A full object oriented code approach will deal with all the information of camera/simulation.

**Data exchange interfaces**

This refers to the Unit's communication interface, which includes all the programming relating with parallelism by using MPI.

**Image rendering**

The environment is currently utilising OpenGL for 3D rendering facilities and can be tested with 100s of eyes within the Swarm Eye Environment.

**Collective image representation**

3D images of a tactical scene and human user interface, and creating a panoramic image to make it easier in understanding the terrain.

**Determine relative positions**

This is a method for determining the relative positions of units in the scene by using the triangularisation technique.

**Moving target detection**

Movement detection module will detect the movement on the scene.

**Self-organised Optimisation**

Position optimisation module will manage the best location and focus of the eye/camera in the scene.

**Tests, Prototype**

A software prototype of a "Swarm Eye". The full testing would be carried out at the end of the software to reduce the errors, software logic etc.

**Report and Discussions**

Project report; Quarterly meetings, delivering the test of the software, any changes which would be essential.

## 5.2 Swarm Eye Software Design

The idea of the Swarm Eye software project is to provide a simulation environment for testing practically various Swarm Eye application setups. The basic principle of Swarm Eye is to provide a set of autonomous eye units which can analyse a tactical scene in parallel, running efficient image analysis in a distributed manner and, thus, providing richer information to the decision maker in the field. The first steps into this direction are now made by providing a parallel communication platform. The project also delivered a parallel scene simulation and analysis platform. The following steps will merge the current development work package strands. After that all the essential steps are made to start with the first distributed image analysis processes.

Software system consisting of:

- Swarm environment
- Swarm Eye units
- Parallel image processing capabilities
- Parallel communication
- User interface

The very first step in developing this “Swarm Eye” software was to design the software and then move onto the implementation. The basic block diagram of the software design is as follows:-

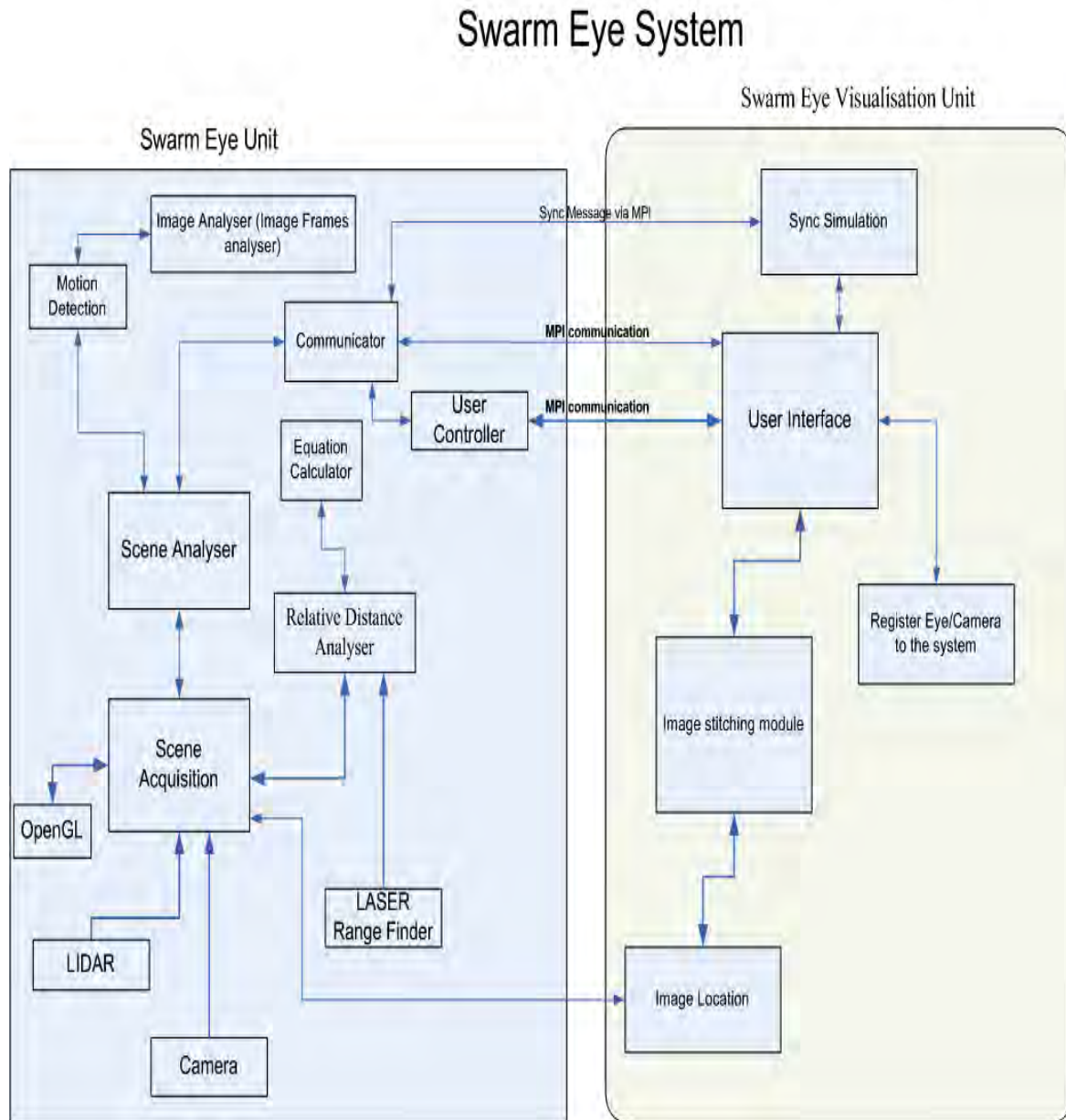
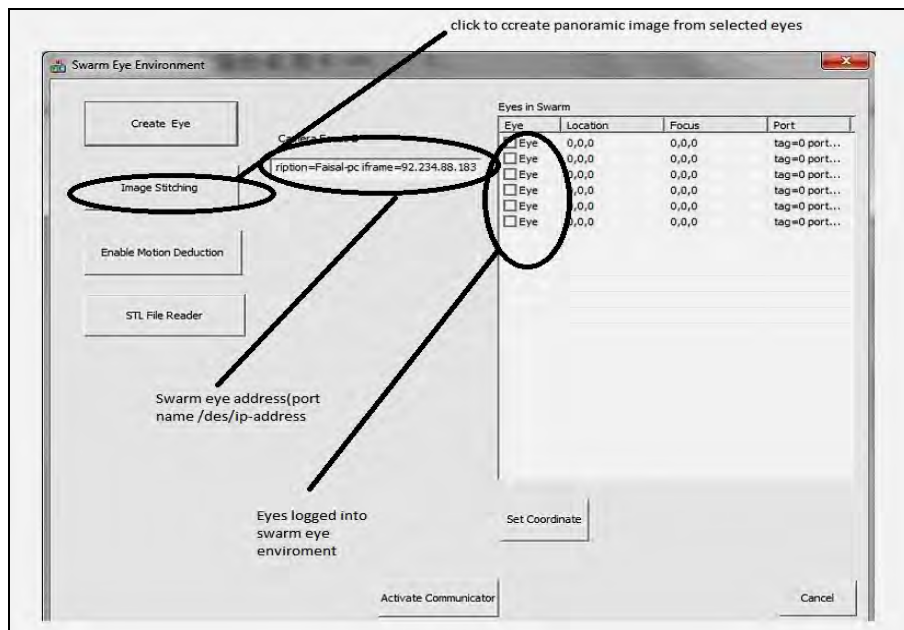


Figure 12: Swarm Eye Software Design

The structure of the software has been developed in an object oriented way. Figure 12 shows the structural layout of the Swarm Eye software. Although the current implementation consists of several individual modules, the system can be divided into two major parts: the Swarm Eye Units and the Swarm Eye User Interface (Visualisation Unit).

### 5.3 Visualisation Unit Design

The Visualisation unit has been designed to monitor all the activity, to issue tasks to the swarm and to provide vital information to the user about the swarm and ground activities. As soon as any UAVs or camera goes airborne it could be added to the Swarm Eye to communicate with other eyes in the sky and become a part of the swarm/group. Users can also change the focus level, change the view angle of the camera, add an eye to the swarm and delete a member from the swarm/group by using Swarm GUI. One can communicate and control the swarm and also individual eyes. Figure 13 shows the GUI of the Swarm Eye Visualisation Unit which consists of Swarm Eye feature activation buttons and an Eye Unit monitoring and control window. The high level commands such as 'Activate Image Stitching' for panoramic imaging is translated into low level commands which are communicated to the individual eyes and executed there.



**Figure 13: Visualisation Unit Graphical User Interface.**  
 (Shows a list of four registered eye units. By ticking the checkboxes you can activate or deactivate individual eyes through the Visualisation Unit.)

The Visualisation Unit (see Figure 13) is a separate object in the Swarm Eye system. It connects to the individual Swarm Eye units via MPI (Data exchange interface). The major purpose of the Visualisation Unit is to link the User with the swarm. Through the Registration module each eye is registered at the Visualisation Unit. This way each eye can be addressed and manipulated independently by the user. Another important feature that affects all registered eyes is the 'Sync' button. The environments are simulated in each Swarm Eye unit independently. This allows for highly efficient communication between the Eye units as only small changes happening in each eye-environment simulation need to be communicated. However, because eyes can be added and run at any time of the simulation, the current status of the environment needs to be synchronised between the eyes. This synchronisation process ensures that all eyes 'see' the same environment with the same dynamics at all times. Dynamic scenes get in-sync and dynamic features of Swarm Eye such as motion detection or panoramic image generation of dynamic scenes become possible. The panoramic image generation module which uses stitching techniques is implemented in the Visualisation Unit. The individual images from the activated Swarm Eye units are collected by the Visualisation Unit and processed centrally. However, it is not necessary to have only one Visualisation Unit. The Unit can log on to any eye in the swarm, thus avoiding a vulnerable hub-spoke configuration where taking out the hub causes a failure of the whole system.

#### 5.4 Camera/Eye Unit Software Design

The Swarm Eye units are the simulation of 'eyes/cameras' of the system. They are programmed as objects so they can be copied and independently used as often as desired by the user. This way it is easy to build a swarm of many similar eye units that can interact with the environment, the user and with each other.

Figure 13 also shows the features that are currently built into the Swarm Eye system. In each Swarm Eye Unit a Scene Analyser interacts with the Scene Acquisition module which interacts with the OpenGL [40] environment, a LIDAR data reader, a camera (for real-world applications), and a Relative Distance Analyser. The Motion Detection module interacts with the Scene Analyser. Each eye unit is linked to a user interface for uploading and downloading information. Whole images or just the changes of the coordinates of the eye units can be relayed. Each eye has a fast communication interface (Communicator) to the Visualisation Unit and the other eyes. The User Controller interprets direct commands from the user for e.g. activating a feature or changing the orientation or position of an eye unit. The eye unit can interpret high level commands and execute them independently. This simplifies the control of the swarm and avoids direct low level (protocol level) interaction between user and the single Swarm Unit. In future applications this module will execute autonomous self-organised Swarm tasks which are initiated by the swarm itself. Each eye will utilise and evaluate the local information accessible by the unit itself thus creating emergent behaviour that helps, for example, to structure the swarm units in a way that

serves e.g. a surveillance task best. At the moment the eyes are placed by default or manually by the user through the User Interface (Visualisation Unit).

Swarm Eye units do not use much built-in intelligence. In fact, the eyes can only take images and process them at a comparatively basic level. This is not a drawback of the system, this is intentional. Swarm systems create emergent behaviour and thus the individual units of the swarm do not need to be very sophisticated. The interactions between the eye units create the desired results. At the moment the eye units can take images and detect motion. The intelligent combination of the eyes and their interaction creates e.g. a panoramic scene or allows for determining the speed of an object from interpreting the motion detection feature of many eyes.

However, the implementation of even those 'basic' features of the Swarm units has already been very challenging. Nevertheless, the Swarm Eye Units have been implemented successfully.

The motivation behind Swarm Eye is to keep the individual eye units and the communication between the units rather basic. As the stitching software is more sophisticated, this software has been implemented into the Visualisation Unit. The stitching protocol arranges the eyes once through a central command and then collects the images from each eye centrally.

Finding an efficient self-organised solution of the swarm structure and efficient distributed image collection is on-going research and yet to be completed.

## 5.5 **The communication structure**

The Swarm Eye consists of a set of autonomously performing eye units. In order to perform the task in a group each individual need to have a fully duplex communication protocol in real time (without significant delay). The messages between the units would be different types of data i.e. Images, simple xyz coordinate and messages /commands. Individuals in swarms interact either directly by communication or indirectly through joint behaviour. In Swarm Eye each eye unit can communicate with each other and the Visualisation Unit via the quasi-standard internet peer-to-peer protocol MPICH2 [22, 23]. All processes in Swarm Eye are running in parallel and can communicate asynchronously. The communication unit also provides a platform for launching new eye units, communication between the objects and to delete the present eye/camera if needed. After each eye is released, it acts as an independent process which runs independently from the initialisation unit. This mimics the independent implementation of each eye unit on an e.g. airborne UAV. The processes which initialised the Swarm will remain intact until it gets the signal to shut down the simulation. A special Monitoring and Visualisation Unit is also generated which observes the current status of each eye. It can control and command each eye in case a central control is desired. In Swarm Eye the port solution is implemented. By using ports each eye unit can uniquely be identified by its port address. Also the termination can be initiated centrally by a shut-down

message to each Eye unit which initiates a well defined shutdown of the whole Swarm Eye system.

Each individual eye unit is designed to be rather primitive. The power of the system comes from an efficient communication and intelligent parallel image processing.

The distributed and parallel image processing approach provides several advantages: it supports the safety of the real system in the field against eavesdropping because all information is distributed within the swarm. As information is distributed in the brain only the correct combination of the right information gives a complete picture. A distributed approach is also more robust against communication problems or damages of single eye units. The swarm can still run properly, even if some units are not fully functional. A parallel approach also helps speeding up the image processing and allows for simulating the physical distribution of the eye units in a real swarm. Therefore, implementing real parallel processing is an essential part of the Swarm Eye project.

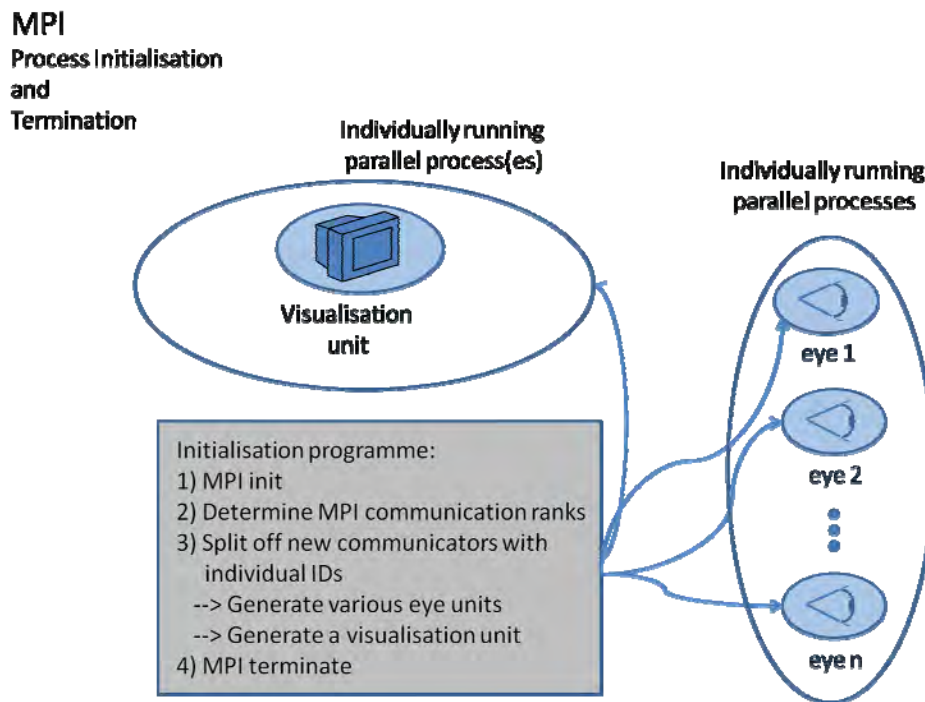


Figure 14: Swarm Eye Distributed Communication

The initialisation programme spawns (or splits off) several processes which are added to the process list of the operation system (Windows or UNIX etc.) as parallel individual processes and can be viewed as processes running e.g. in the task manager under Windows. Figure 14 also illustrates the concept of the communication within Swarm Eye. During the start up of the system the communication module is initialised and several Swarm Eye units are established. Also the Visualisation Unit is set up and displayed. In Swarm Eye new units can be added dynamically thus the communication links are established on demand. During the



run of Swarm Eye all communications are realised through MPICH2. MPICH2 provides a flexible and comparatively easy to implement programming library that can be used for message passing between ports or internet IP addresses.

The current implementation fully supports the notion of having parallel and autonomous eye units. Each eye unit holds a representation of the current environment. This mimics the data feed provided by a camera mounted on a UAV. To support multi angle viewing the environment is modeled using the OpenGL 3D library. This allows for modeling of complex 3D tactical scene to be simulated and rendered in 2D frames for further image processing.

The initialisation unit, the Visualisation Unit as well as the eye units is modularly implemented in an object oriented manner. The communication between the initialisation unit, the Visualisation Unit and the eyes is based on MPICH2 via message passing. A message protocol handler utilises MPICH2 APIs for communication between the units. All communication is port based, thus allowing for parallel execution of these units on individual computers.

The port based communication model also allows for scalability as multiple numbers of eyes can be added to the system. Each eye has its own environment, its own view angle and position in space. The eye units run in parallel together with the Visualisation Unit. For proof of concept only one Visualisation Unit is realised. At the moment the eye units coordinate with each other and the visualisation unit via custom messages.

## 5.6 Virtual environment

The virtual environment simulation has been created to simulate the camera (eye) as shown in figure 15. This is a 3D simulation which will allow placing of camera or eye at any desired location in the scene. Simulation is cost and time effective as compared to a real camera. There are a few simulations which have been developed and have been adopted. The simulation has been plugged into the GUI as a module and can be replaced by different simulations or by real cameras.

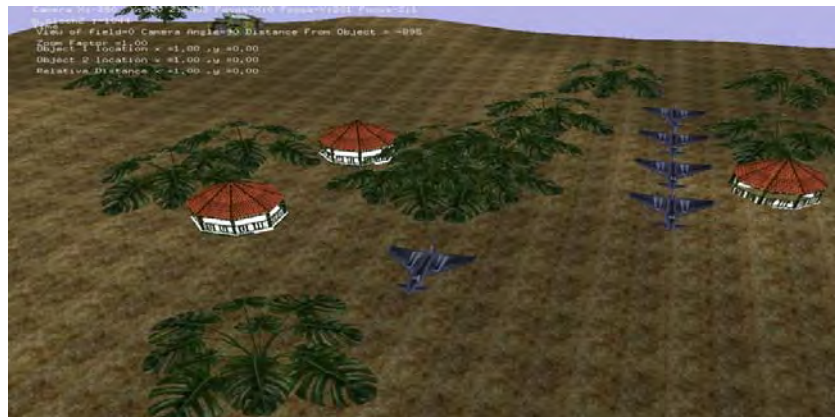


Figure 15: The scene generated by using md2 object.

There are few simulations that have been created from the start of the project which are all developed in C++/C and by using OpenGL. The C++/C offers to be very effective in processing consumption which is very crucial for this project.

## 5.7 Image Rendering Package

The image rendering module is a key module in Swarm Eye. It supplies the eye units with images from the simulated environment. Significant time was spent in the project in developing an image rendering package that provides simulations that looks as realistic as possible. Today the image rendering module is the principle platform for experiments in Swarm Eye. All following experiments on collective image representation, position determination and movement detection have been implemented using this module.

The image rendering engine used in Swarm Eye uses the MD2 [41] file format and the OpenGL library for image rendering. During the project the image rendering module in Swarm Eye was significantly modified and improved from a very basic triangulation viewer to a very sophisticated 3D rendering engine as shown in Figure 15

Each Swarm Eye unit can observe a scene from any point in space with any view angle. Figure 16 shows the scene seen by four different eye units.



Figure 16: Views as seen by four independently running Swarm Eye units.

Each eye unit can fly through a scene following a specified trajectory. The images as seen by each individual eye are displayed at the Visualisation unit in separate windows. In the example in Figure 16 also additional monitoring information is displayed in head-up mode

inside the windows. The small car that can be clearly seen in the left bottom image moves dynamically inside the scene and its movements are displayed dynamically.

From the example above one can also see that Swarm Eye can provide useful information about hidden objects or moving objects. In the top left image the car is not visible while it is in the other three images. This very simple example illustrates that a swarm is capable of gathering more information than a single camera. In combination with the motion detection capability this feature of the swarm can become very powerful. This allows for elimination of blind spot limitations associated with a single camera. Static single camera systems can also not see behind the obstacles. Also, the determination of the direction of movement becomes much easier to perceive when inspecting a scene from multiple angles.

## 5.8 Collective Image Representation

Swarm Eye is a concept which can be applied to many areas. The idea of Swarm Eye is to use a set of observing eyes in parallel to generate more valuable information than possible with single camera solutions. The aim is to provide a decision maker with more valuable information in critical tactical situations.

### 5.8.1 Panoramic image generation

In Swarm Eye the problem of generating panoramic images has been solved by image stitching. Image stitching is the process of combining several images together to form one single new image showing the content of the individual images in one single picture. The technology of image stitching is rather advanced today [10, 11, and 12]. Swarm Eye makes explicit use of these technologies by incorporating them into the system.

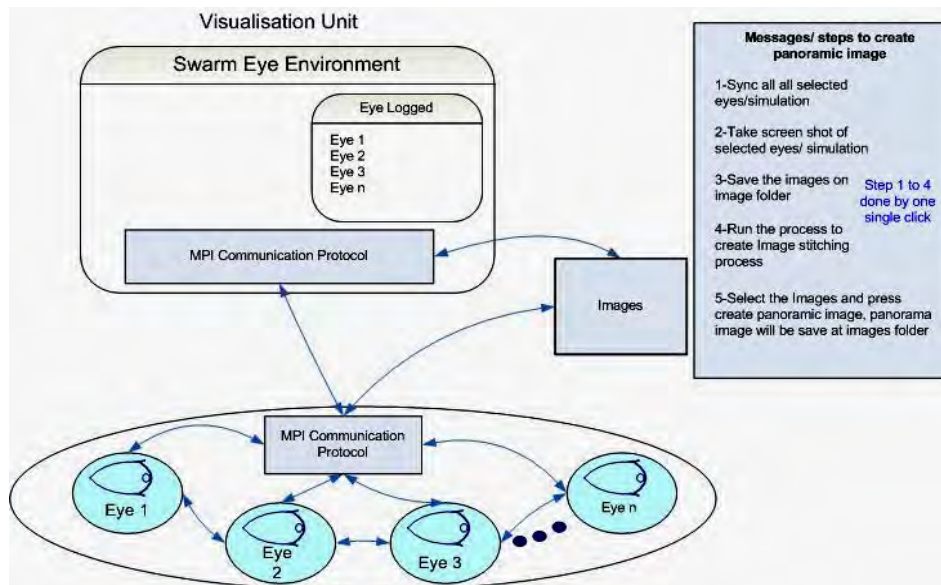


Figure 17: Panoramic view generation in Swarm Eye.

Figure 17 shows the general structure of the panoramic image generation process. The eye units contributing to the panoramic image generation are first selected in the Visualisation Unit by the user. The scene is synchronised via MPICH2 message so that every eye has the same time frame. In real-world scenarios the synchronisation procedure is, of course, not necessary. Each eye then takes a snapshot of the scene and sends it to the Visualisation Unit. In order to perform image stitching in Swarm Eye, the individual views of the eye units need to be synced and offer overlapping coverage of the scene. In practice any orientation of the eyes can be used for this purpose as long as there is a certain amount of overlap between the individual images. An overlap of ca 20% between each image is generally recommended to allow the stitching software to identify unique markers within the overlap area to determine stitching points. The marking procedure is performed by the Scale-Invariant Feature Transform algorithm SIFT [13]. The RANSAC (RANdom SAmple Consensus) [14] algorithm is used to compare similarities between the point sets identified by SIFT in the individual overlaps. The images are adjusted and transformed so that similar point clusters overlap with the highest probability. The adjusted images are then rendered and combined into one single image. In Swarm Eye the images are collected and passed to the professional image stitching software Autostitch<sup>®</sup> [15, 16]. This software is embedded in Swarm Eye. However, it can be replaced by any other automated stitching software if necessary.

In Figure 18 the views of four different eye units are displayed showing a landscape with trees, two houses and some jet planes sitting on the ground. One can see that each eye is pointing in a slightly different direction covering the major area of a scene.

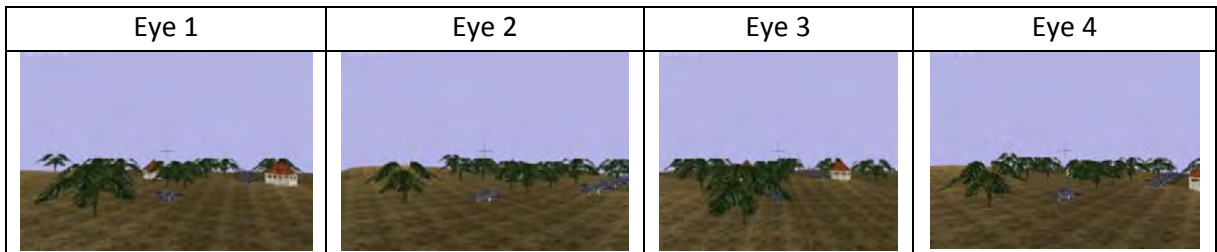


Figure 18: Views of 4 different eye units covering a tactical scene.



Figure 19: Panoramic image created by combining the images from Figure 18.



Figure 19 shows the compiled panoramic view. After activating the “Image Stitching” button in the Visualisation Unit (see Figure 13), this image can be seen. All four images are sent by the eye unit to a shared common port. From there the stitching software automatically collects the images and joins them to a single panoramic image. The image is displayed automatically by Swarm Eye. A copy of the image is placed in the file system of the operating system and can be browsed by any conventional image viewer.

### 5.8.2 Swarm Formation and Panoramic Images

Swarm inherently operate in a particular formation. The position and orientation of an individual swarm member within a formation affects its individual view and the overall stitched panorama. Different swarm formation coverage of a particular area of interest may yield a different type of panoramic view. This section presents some interesting panoramas created by stitching individual views of Swarm Eye units in different formations.

The panorama in Figure 20 is created by stitching individual views of swarm units in curve formation.



Figure 20: Panoramic View Yielded by swarm in curve formation

The panorama in figure 20 is based on individual views of 15 swarm units. The formation and orientation of individual swarm is presented in figure 21. The individual views are provided in Annex A.

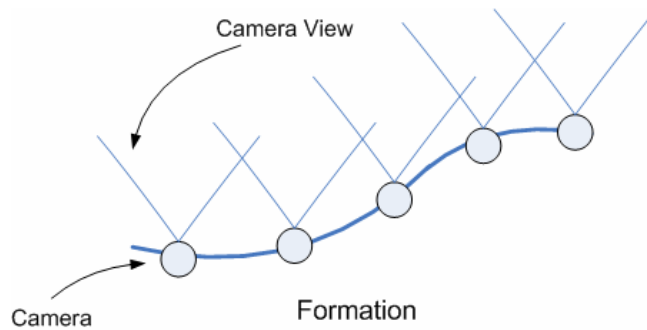


Figure 21: The orientation of swarm units in curve formation

No of Input Images	Resolution of Individual Image	Resolution of Panorama	Panorama Tool Used
13	1024x796	1756x850	Microsoft ICE

The collaborative viewing capability of swarm allows for coverage of wider areas for an aerial surveillance perspective. A downward camera orientation within an arrow formation can be used to generate the collaborated view of a wider area. This example is presented in figure 22.



Figure 22: Wider area coverage by swarm in arrow formation

No of Input Images	Resolution of Individual Image	Resolution of Panorama	Panorama Tool Used
26	1024 x 696	1400 x 727	Microsoft ICE

Figure 22 was generated by the swarm in arrow formation with a downward camera orientation towards the ground. This picture provides the vital information of the ground beneath the aircrafts.

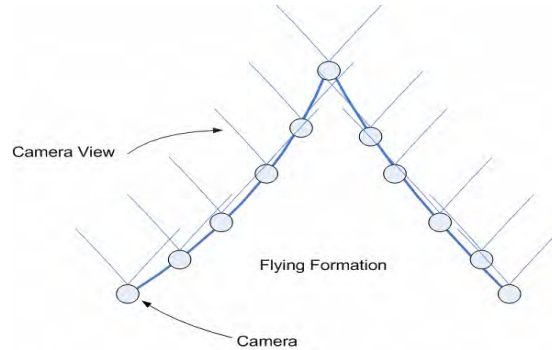


Figure 23: Swarm in arrow formation with downward camera orientation

Several types of UAV can stand still in air allowing them to form formations that are usually not possible with wing based UAV. For instance rotorcraft can rotate in the air and this allows them to capture 360 coverage of the area of surveillance. Furthermore, the flying capability allows them to form a circular formation. Figure 23 discusses a usual 360 degree panorama created by swarm in a circular formation.



Figure 24: The circle formation of panoramic image with viewing outside the circle camera orientation

No of Input Images	Resolution of Individual Image	Resolution of Panorama	Panorama Tool Used
13	1024x796	1756x850	Microsoft ICE

The images were taken in circular formation with all swarms focusing towards the outer of circle.

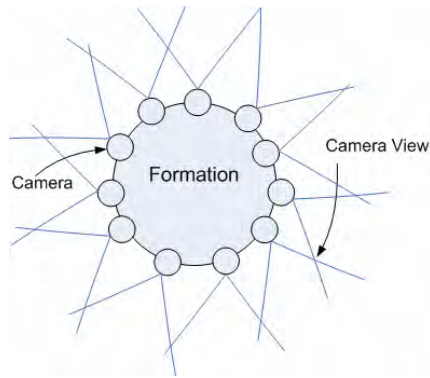


Figure 25: **Swarm in circle formation focusing towards outer circle**

Interestingly it is also possible to change the camera orientation within a circular formation to focus towards the centre of the circle.

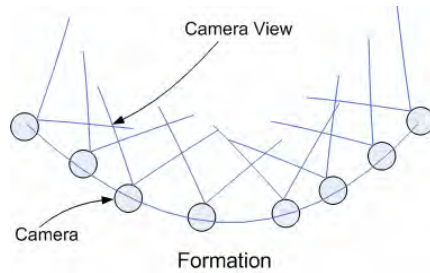


Figure 26: **Swarm in semi circle formation focusing towards inner circle**

This allows for circular angular coverage of an area of surveillance and stitching individual views can generate spherical panoramas. The semi circle formation shown in figure 26 allows for multi angle coverage of the surveillance area. A full circle formation can also aid in acquisition of 3D information about the object of interest within a scene.





Figure 27: Swarm Circular formation with all swarms focusing towards the centre of circle

No of Input Images	Resolution of Individual Image	Resolution of Panorama	Panorama Tool Used
14	1024x696	914x887	Microsoft ICE

The quality of generated panorama is dependent on several factors such as overlap, view orientation and the algorithm used to stitch images together. In certain instances a particular algorithm may fail to generate a panorama if any of the required parameters of the algorithm are not met.

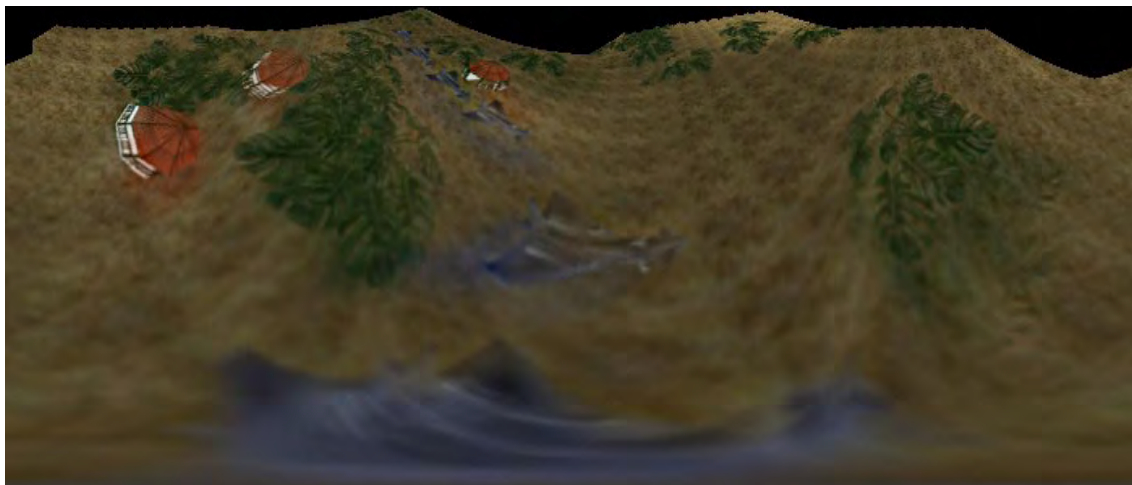


Figure 28: A distorted panorama generated using SIFT and RANSAC

No of Input Images	Resolution of Individual Image	Resolution of Panorama	Panorama Tool Used
14	1024x696	1400x310	AUTOSTITCH

The distorted panorama in figure 28 was created by SIFT using the same set of images used to create the panorama in figure 23. However, the panorama in figure 27 was created by using Microsoft ICE.

## 5.9 Determine Relative Positions

The identification of Position determination without GPS is a navigation task widely known for centuries by sailors, travellers and pilots. Today's most of the UAVs position establishment is mainly based on DGPS/GPS which is very useful and precise. However, relying on GPS/DGPS only could cause great difficulties in case GPS/DGPS is jammed. Also a precise position identification of objects within a scene is of high interest for example in object tracking. GPS will give you some indications about the positions of yourself or fixed objects in the scene but generally no information about the relative position of an unknown object one is targeting in a scene.

The few options for relative position determination have been tested for Swarm Eye. The first technique determines the position of an observer within a scene. The second determines the position of a targeted object within a scene. The third option generates a global map of multiple positions at the same time using several eyes. In the above approaches the object boundaries have been hypotheses in the image.

### 5.9.1 Determination of the observer position within a scene

This approach for determining the position of the observer/camera/eye within a scene uses a classic navigation technique called triangulation or resection. A typical approach to position detection is taking visual information and globally available static position data. Examples for static data are a compass or the stars. Visual information can be compared with maps. Depending on the unknown and known parameters such as angles and distances, one can determine the position of objects rather precisely. However, the Swarm Eye approach has an advantage above the conventional triangulation because here many eyes can be used to determine a position making the calculations even more accurate.

The determination of the unknown position of an observer needs two given positions of observable objects and two angles of the directions relative to a north-south orientation. The determination of the angles can be done using a compass. The data of the position of the two objects may come from a map. The calculation of the unknown position  $(x_1, y_1)$  from the given positions  $(x_2, y_2)$  and  $(x_3, y_3)$  and the angles  $\alpha$  and  $\beta$  (see Figure 29) is as follows:

$$x_1 = x_2 + (y_2 + (x_3 - x_2 - y_2 \tan \alpha - y_3 \tan \beta) / (\tan \alpha + \tan \beta)) \tan \alpha \quad \text{Equation (1)}$$

$$y_1 = (-x_3 + x_2 + y_2 \tan \alpha + y_3 \tan \beta) / (\tan \alpha + \tan \beta)$$

Equation (2)

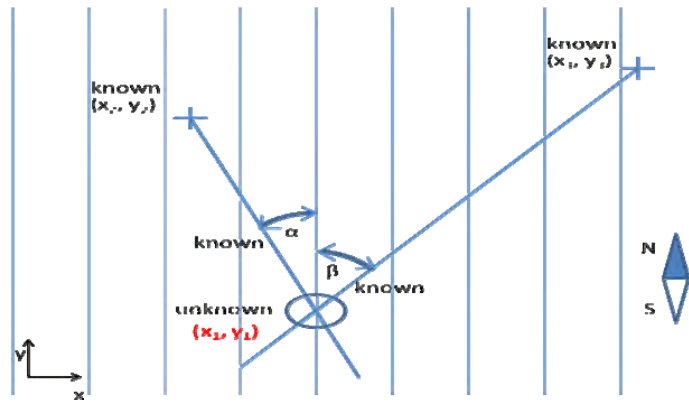


Figure 29:: **Triangulation (resection) using a single eye to determine the position  $(x_1, y_1)$  of the eye itself given two positions  $(x_2, y_2)$  and  $(x_3, y_3)$ , respectively, and the angles  $\alpha$  and  $\beta$  relative to a north-south orientation on a map.**

In the Swarm Eye simulation environment the implementation of this technique looks as shown in Figure 30. The position determination uses formulas 1 and 2. Each view angle  $\alpha$  and  $\beta$  is realised by an individual eye unit placed at the same place as the observer. The positions of the tips of the two houses, i.e.  $(x_2, y_2)$  and  $(x_3, y_3)$ , are given. The angles  $\alpha$  and  $\beta$  can be determined by rotating the eye units  $\alpha$  degrees left and  $\beta$  degrees right, respectively, focusing the tips of the gazebos.



Figure 30: **Determining the individual position by triangulation. North-south directions are indicated by the blue-white pointers. The angles  $\alpha$  and  $\beta$  as well as the positions of the houses are known.**

In this example, the north-south direction is known in Swarm Eye. Having this data is a realistic assumption as electronic compasses are readily and cheaply available. In Figure 30 the blue tip of the compass rose indicates north. In Swarm Eye the position and the view angles are constantly monitored by the Visualisation Unit via MPICH2. This allows for an

instantaneous automatic calculation of the observer position. This information is displayed in the Visualisation Unit.

### 5.9.2 Determination of a target position within a scene

Formulas (1) and (2) can also be used to determine the position of an unknown target object. Here, knowing the position and view angles of two individual eyes targeting the object is sufficient.

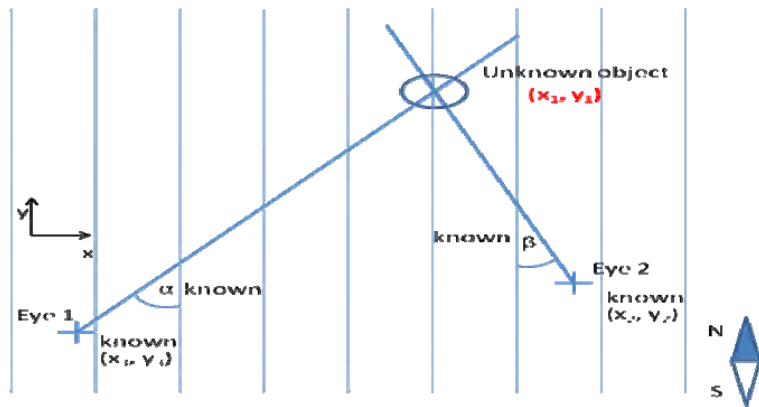


Figure 31: Determining an unknown position of a third object by triangulation. North-south directions are indicated by the blue-white pointers. The angles  $\alpha$  and  $\beta$  as well as the positions of the eyes 1 and 2 are known.

The same formulas as in the previous position determination procedure can be employed except that the eyes are now placed at two different known positions  $(x_2, y_2)$  and  $(x_3, y_3)$ . The eye units 1 and 2 are now targeting an unknown position of an object  $(x_1, y_1)$ . The view angles  $\alpha$  and  $\beta$  are the same as in Figure 31 but associated to the individual views of the eyes. Figure 32 shows an application of the triangulation for unknown object position detection. In the figure the position of the jet and the position of a Swarm Eye (indicated by the cross in the middle of the photo) are known. Both objects have visual access to the small green four-by-four car hidden behind the palm tree in the bottom right side of the image. They can follow the dynamic movement of the car and track its position online. The positions are displayed (upper left corner in Figure 29) at the Visualisation Unit. In this example the Visualisation Unit uses another Swarm Eye unit to observe the whole scene. This demonstrates that Swarm Eye can make use of any Eye unit to get a full view of a scene while collecting vital data from different collaborating Eye Units.

The car in the scene in Figure 32 is moving. Object position determination using image data for speed determination is still an issue with current consumer based devices although first attempts to change this with single camera systems were just recently published [18]. As shown in Figure 32 this can be overcome more easily with Swarm Eye. Automatic object recognition is today rather advanced and readily available. The modular structure of the Swarm Eye software makes it possible to integrate those features into Swarm Eye in the future.





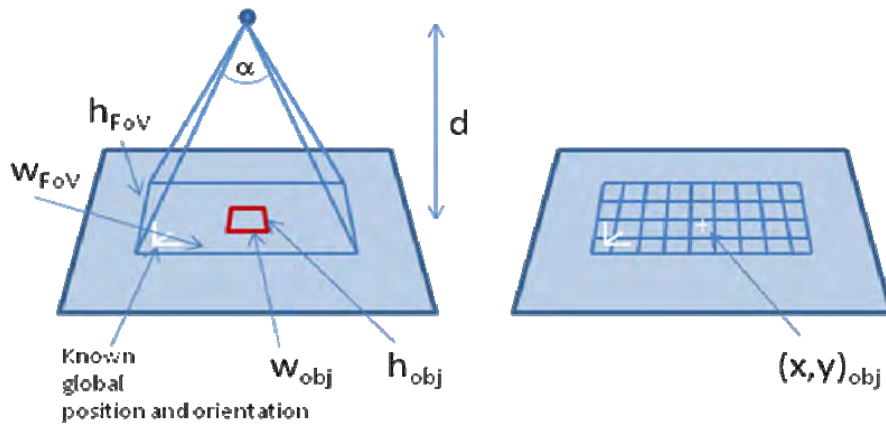


Figure 33: Global positioning using information about the Field of View (FoV). Left: calculation of the dimensions of an object by using at the relative proportion of the dimensions of the object within the FoV. Right: coordinate system with relative positions of points at the grid crossings.

Tilting the view angle changes the length of the field of view. This can be compensated by adjusting e.g. the grid structure (see Figure 32 right) accordingly. The adjustment follows the tangent formula taking the inclination angle into account which is known by the Eye unit.

Figure 34 shows the application of the proportionate dimension technique in Swarm Eye applied to the determination of the position and dimension of unknown objects. The global dimensions of the scene are calculated by taking the FoV information into account (large green arrows). The position of the jet in the back of the image as well as the length of the first jet is determined by comparison with the grid dimensions plotted at the border of the image which takes the tilt of the view angle already into account.

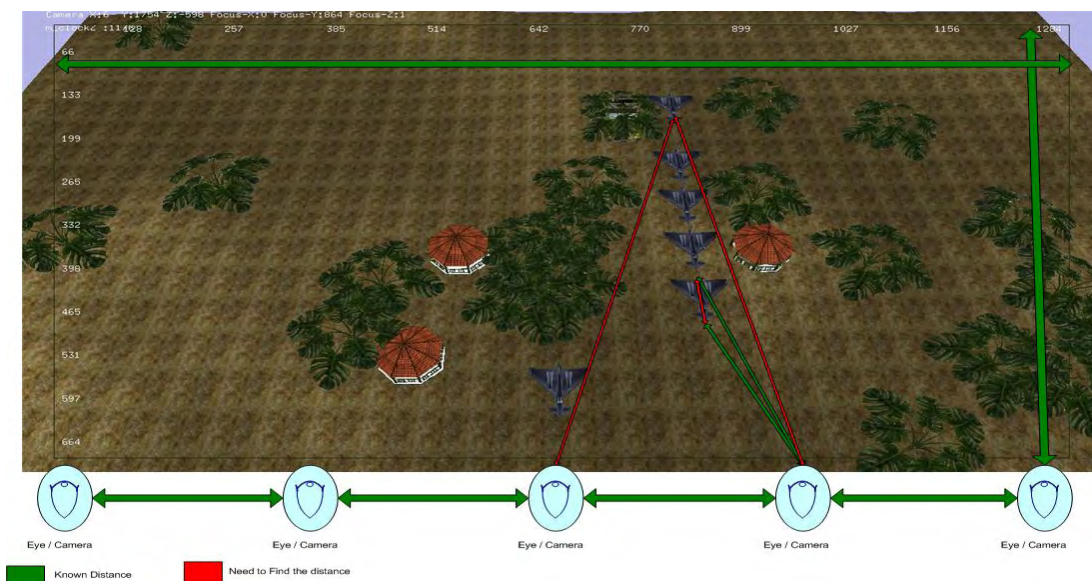


Figure 34: Determination of the position and dimension of objects in Swarm Eye by proportionate dimension calculations.

### 5.9.3 Moving Target Detection

There are various motion detection algorithms in the field of computer vision. There are mainly two different scenarios for motion detection for Swarm Eye project i.e. Stationary camera detecting motion of moving objects and moving camera (for example camera mounted on flying UAV/aircraft) detecting motion of moving objects. Today's numerous UAVs, CCTV and other surveillance and security equipment uses motion detection sensors. However motion detection sensors can lead to very complex image analysis by exploiting the Swarm Eye concept. This is motivated by the so called 'flicker effect' which can be observed from insects with compound eyes. Each element of the composite eye can capture only a very coarse and partial image of the world. However, the combination of many eyes into one compound eye allows the insect to register even the slightest movements within a very large field of view and even determine the speed and size of the object. Image analysis of real-world footage is rather complex.

To demonstrate the power of the Swarm Eye concept, real-world image motion detection has been tested. The open source motion detection software (Motion Detector V.1.50 by Andrew Kirillow) has been used in this Swarm Eye project.

### 5.9.4 Swarm Formation

The formation algorithm has two parts in the software development. The first part is to form the initial formation and second part is to form a dynamic formation when a blind spot is being hit or the required object is not visible.



Figure 35: The user interface allows for selection of different formations and swarm units.

The figure 35 shows the interface which has been designed to analyse different formations and the designed algorithm in different scenarios. The radio button shows how the different formations will capture the visual data. The user can change the number of swarm units in the simulation. The auto option could be selected if the user intends to see the dynamic algorithm used and tested. The implementation of dynamic algorithm has not been completed and is discussed under the future work section.

The selected formation is applied at the initial stage of the swarm lifecycle. If during the simulation a blind spot is encountered by a swarm unit then this event triggers a dynamic reformation of the swarm.







Figure 37: **The Line formation of the swarm units, where each aircraft demonstrates a swarm unit.**

Individual swarm units have a limited area of observation. However, when multiple swarm units are placed in an appropriate formation, this yields a very vast collaborated area of coverage. To maintain the optimum collaborated coverage of the area of surveillance, the individual swarm units maintain a sight oriented distance between them.

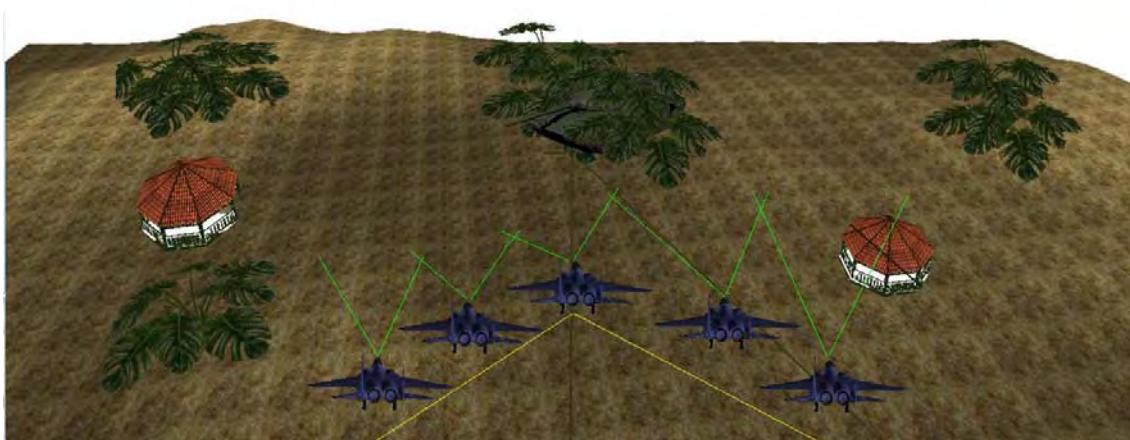
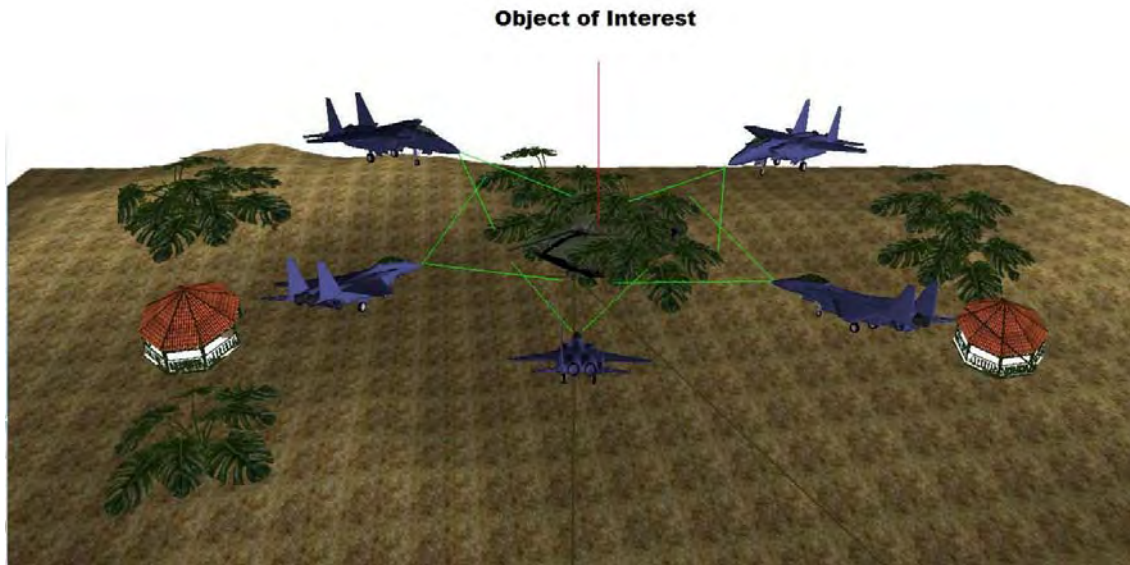


Figure 38 : **The Arrow formation of the swarm units.**

Figure 38 presents this scenario where 5 swarm units are in an arrow formation with the whole swarm coverage area. This feature allows for wide range panoramic views of the scene of interest to be generated.

However, the generation of 3D images from 2D captured images requires suitable multi-angled 2D images. This is where dynamic formation plays a vital role, upon identifying a suitable object of interest; the swarm may initiate dynamic formation. The formation will consider the number of units in the swarm, the requirement to cover the object of interest from various angles and then apply a suitable formation to the swarm. This scenario is presented in the Figure 39, where the tank acts as the object of interest and has been covered by the various units in the swarms from different angles.



**Figure 39: Dynamic Formation Figure and their explanation**

These dynamic formations are essentially objective driven. The objective may vary from multi angle surveillance of a specific object of interest to acquisition of sufficient multi angle data to construct a 3D representation of a tactical scene. This scenario is observed in the transformation of formation from figure 38 to Figure 39.

### **5.9.5 Transformation Steps**

The transformation is governed by the following selected steps of the algorithm.

Step 1: The required object within the tactical scene is scanned by each individual eye. This will generate the visual index of the individual eyes.

Step 2: The Unit with the highest visual index and closest to the required object will subtract the background from the object and then detect the centre point of the visible required object.

Step 3: The centre point of the required visible object is calculated and the distance from the centre point to a blind spot of the object is established.

Step 4: The calculations above will allow for the selection of swarm units for repositioning in order to increase their visual index factor and the overall collaborated visual coverage of the area.

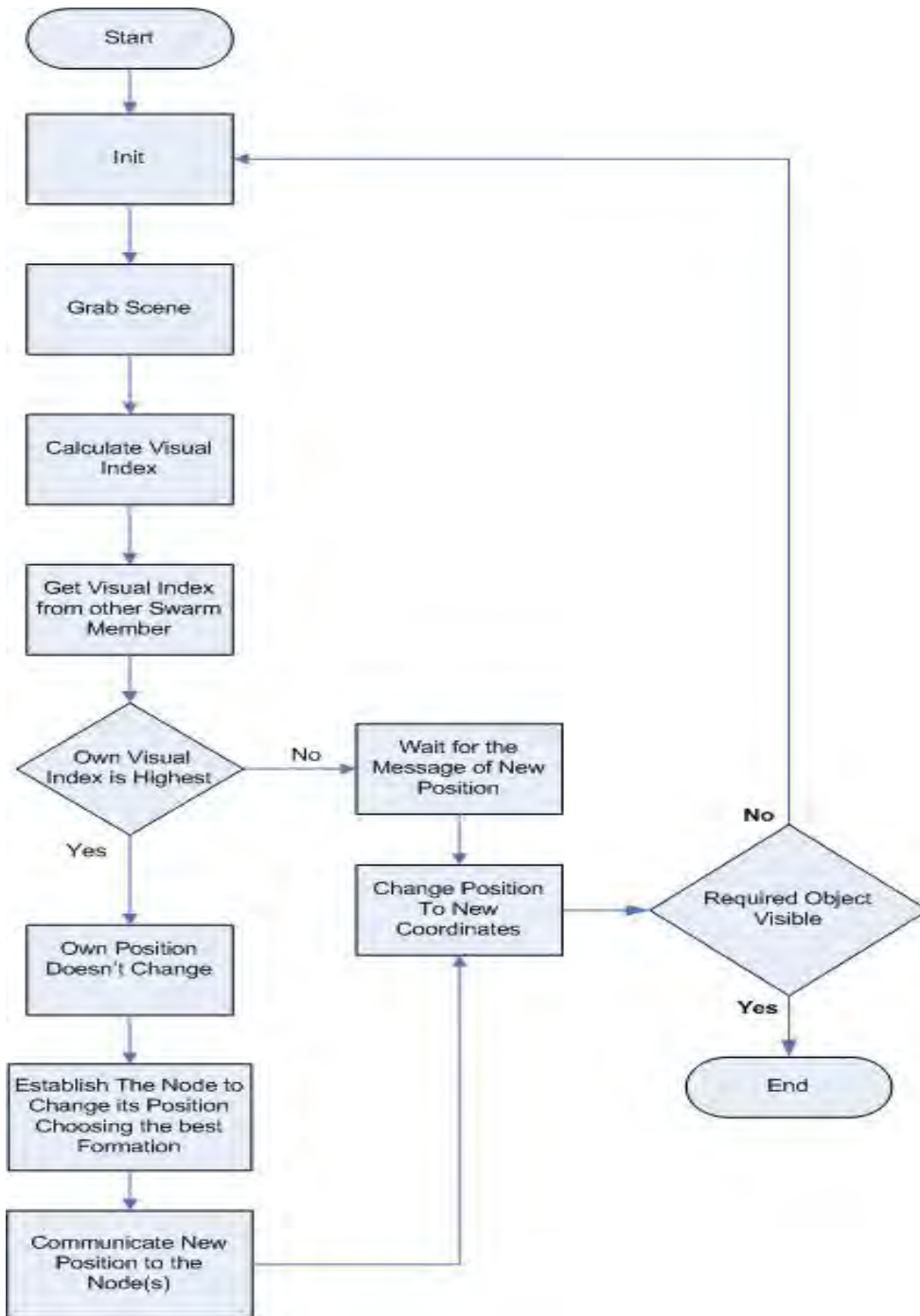


Figure 40 Block Diagram of Dynamic Formation Algorithm

The steps discussed in the diagram above have been implemented as communication between various classes within the system. The communication steps have not been highlighted in Figure 40.

The UML sequence diagram for this case is discussed as under.

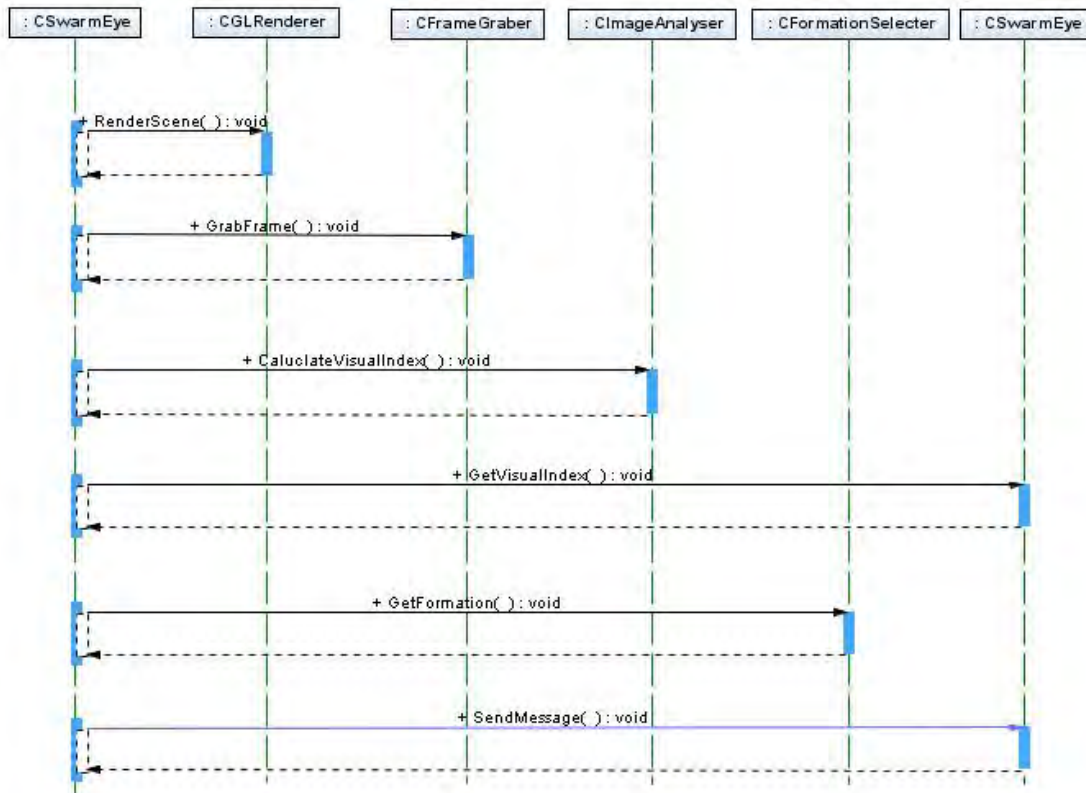


Figure 41: Dynamic Formation Sequence Diagram of basic objects

Figure 41 shows the basic classes and their messages to obtain the formation, based on the visual index.

### 5.10 UAV Physical Manoeuvring Constraints

An important aspect affecting UAV manoeuvring capabilities is the physical constrains. These constraints vary based on the model of the UAV. Most UAV can be distinguished as either fixed wing based or rotorcraft. The manoeuvring of the UAV has physical constraints interims of airspeed, turn rate, control response latency and aerodynamics effect on the orientation of the UAV [45]. These manoeuvring capabilities play a vital role in selection of a UAV for a particular mission.

Rotorcraft based UAV can stay stationary in the air, while the wing based UAV cannot. Furthermore, the take off and landing capabilities greatly differs in both wing based and rotorcraft UAV. The rotorcrafts can also manoeuvre at 360°.

The Swarm Eye system allows for both rotorcraft and wing based UAV flight simulation. The current implementation of the swarm formation algorithm does not consider all the physical

constraints of wing based UAV. However, it assumes that any UAV can move to a certain position in relation to the particular swarm formation. It is assumed that all manoeuvring physical constraints are met by UAV and there are no worse case scenarios.

The reason behind this assumption is that UAV physical manoeuvring constraints are beyond the scope of this project.

### 5.11 Present Work

The concept of swarm and its application in tactical decision making has been demonstrated by the implementation of the Swarm Eye project. The following software features have been implemented:

- Virtual environment
- Software structure
- Data exchange interface
- Image rendering / visualisation
- Collective representation of a scene (panoramic view)
- Relative position identification
- Motion detection

The proposed Formation Algorithm has been partially implemented and requires optimisation.

### 5.12 Implementation Performance Analysis

The Swarm Eye implementation test bed configurations are listed in the table below:-

Description	Value
OS	Windows7 Pro
CPU	Intel iCore7 720QM @1.6Ghz (x4)
RAM	4GB (DDR3 Dual Channel)
MPICH2	Verson2.0 Argonne

The test bed comprised of an Intel Icore7 processor with hyper threading technology. This allowed the Windows 7 operating system to observe a total of 8 virtual cores. Maximum per CPU core load test was conducted to establish CPU performance and memory consumption overheads. The results are listed in the following table.



<b>Simulation Instance</b>	<b>Memory (KB)</b>	<b>CPU Usage %</b>	<b>Simulation Status</b>
<b>1</b>	<b>31,896</b>	<b>16%</b>	<b>OK</b>
<b>2</b>	<b>31,844</b>	<b>18%</b>	<b>OK</b>
<b>3</b>	<b>32,020</b>	<b>17%</b>	<b>OK</b>
<b>4</b>	<b>31,624</b>	<b>21%</b>	<b>OK</b>
<b>5</b>	<b>31,816</b>	<b>20%</b>	<b>OK</b>
<b>6</b>	<b>31,764</b>	<b>18%</b>	<b>OK</b>
<b>7</b>	<b>31,996</b>	<b>19%</b>	<b>OK</b>
<b>8</b>	<b>31,636</b>	<b>22%</b>	<b>OK</b>

The results confirm that the simulation platform can successfully scale to a higher number of swarm units on a single machine without suffering from any performance degradation. Furthermore, in a network based environment a single swarm unit can be hosted on an individual machine and this approach allows the simulation to scale beyond the limitations associated with a single host. However, message between these swarm units is subject to bandwidth limitations associated with network infrastructure.

### **5.13 Future Work**

The thesis has presented a novel approach for the swarm formation based on VFI. This approach allows dynamic swarm formations with the focus of maximum coverage of an object of interest. This approach can aid in tactical decision making and allows for autonomous swarm behaviour.

Swarm Eye was a one year project. For an optimised implementation of the swarm formation algorithm this project would have to be continued to PhD level. The directions of future work are proposed as under.

- The initial formation algorithm has been implemented but the dynamic algorithm implementation is to be undertaken.

- The optimised implementation of the swarm formation algorithm with the integration of other collaboration algorithms such as (collision avoidance, obstacle avoidance, target seeking and formation keeping) at present the Swarm Eye system does not fully implement these methods.
- Demonstration of the results in different scenarios to perform analysis of the swarm formation algorithm.
- Implementation of image analysis by swarm unit to help tracking, detecting the target and form dynamic formation by applying the swarm formation algorithm.
- The analysis of real time constraints and time delay of the Swarm Eye formation building and processing of the visual data by the swarm units could be implemented and tested on real units rather than simulation.
- The UAV physical manoeuvring constraints to be catered by the algorithm in more detail.

The Swarm formation is a considerably new topic and minute research has been performed. The area of swarm formation, coordination of multiple autonomous flying vehicles offers diverse edges of Swarm and still need further exploring and development.

The Swarm Eye project is open to many applications; this project provides the foundation of the Swarm Eye concept and could also benefit different industries.

## 6 Conclusion

---

Distributed ground teams or swarms of UAVs can provide different and dynamic views of a tactical scene. Swarms are sets of interconnected units that can be arranged and coordinated in any flexible way to execute a specific task in a distributed manner.

Swarm Eye provides a simulation environment to explore the potential of swarm and its merger with various advanced technologies to aid in surveillance and reconnaissance missions.

The identification and implementation of the proper formation of a swarm is a vital element for efficient swarm application. The formation is derived by a suitable algorithm that is applied on the swarm members to achieve a particular orientation of these members within the swarm.

The initial formation refers to the formation which will be adopted by the swarm at the start up. As soon as the unit will be airborne and get registered into the swarm they will form a formation to fly. The dynamic formation will be adopted by the swarm if the required objects are not visible due to the blind spots in the current formation.

In this report, a formation algorithm has been proposed which demonstrates that blind spots can be minimised to get a better view of the target object or terrain. This formation algorithm aims to derive a suitable formation based on the *visibility factor index*. This abstract term refers to a quantifiable unit that can measure visibility with regards to a particular *object of interest* within an area of surveillance. A higher visibility of the *object of interest* yields a higher visibility factor index. This quantified visibility unit is then used in an algorithm to apply a suitable formation to the swarm units.

The objective of the algorithm is to apply a suitable formation to the swarm so that maximum collaborated visibility is achieved. The algorithm divides a swarm into several sub-swarms applying suitable formations at both the sub-swarm and the overall swarm level. This approach allows for individual sub-swarms within a main-swarm to have their own formation. The division of a whole swarm into sub-swarms is very effective, by having sub swarms the user can allocate different jobs to different groups and sub swarms are easy to manage as compared to managing whole swarms with the units of few hundreds.

The Visualisation Unit is a separate object in the Swarm Eye system. It connects to the individual Swarm Eye units via MPI (Data exchange interface). The major purpose of the Visualisation Unit is to link the user with the swarm. Through the registration module each eye is registered at the Visualisation Unit. This way each eye can be addressed and manipulated independently by the user.



The Swarm Eye units are the simulation of 'eyes/cameras' of the system. They are programmed as objects so they can be copied and independently used as often as desired by the user. This way it is easy to build a swarm of many similar eye units that can interact with the environment, the user and with each other.

The Swarm Eye consists of a set of independently performing eye units. In order to perform the task in a group each individual needs to have a suitable communication protocol to allow for communication between the units. The messages between the units would be different types of data i.e. Images, simple 3D coordinate and messages /commands. Individuals in swarms interact either directly by communication or indirectly through joint behaviour. This coordinated behaviour allows for deployment and experimentation with the swarm of variable sizes. The simulation platform produced by this work provides an extensible framework for further exploration and experimentation with the autonomous swarm operations and formations with regards to tactical decision making. The self formation capability using Visual Field Index allows for autonomous swarm formation and aids surveillance with reference to a specific object of interest. The work can be further extended to identify suitable formations for a particular scenario of interest and impact of different scene conditions such as light, weather and terrain topologies. It further allows for the evaluation of the impact of message delays and optimised image analysis techniques to cater for a particular scenario. Overall, the work presented in the thesis sets up the background for further PhD level work into the fields of autonomous swarms, optimised swarm formations, communications and image processing techniques for swarm intelligence.

## 7 References

---

- [1] G. Swarup, S. Ananthakrishnan *et al.* "The Giant Metre-wave Radio Telescope", *Current Science*, vol. 60, No. 2, pp. 95-105, 1991.
- [2] P. Best, "LOFAR-UK", Technical report, 2007 (<http://www.lofar-uk.org/>).
- [3] J. Howard, "Practical HDRI: High Dynamic Range Imaging for Photographers", Rocky Nook, 2008.
- [4] J. Gulbins, R. Gulbins, "Photographic Multishot Techniques: High Dynamic Range, Super-Resolution, Extended Depth of Field, Stitching", Rocky Nook, 2009.
- [5] S. Miller, "All Kinds of Eyes", Marshall Cavendish, 2007.
- [6] G. Struwe, "Spectral Sensitivity of the Compound Eye in a Moth. Intra-and Extracellular Recordings", *Acta Physiologica Scandinavica*, vol. 87, No 1, pp. 63 – 68, 2008.
- [7] L. Bishop, "An ultraviolet photoreceptor in a Dipteran compound eye", *J. comp. Physiol.*, 91, 267-275, 1974.
- [8] E. Buschbeck, B. Ehmer, R. Hoy, "Chunk versus Point Sampling: Visual Imaging in a Small Insect", *Science*, Vol. 286. no. 5442, pp. 1178 – 1180, 1999.
- [9] R. Koshy, R. Munasinghe, A. Davari, "A Design of a Composite Eye for Computer Vision," *System Theory*, 2008. SSST 2008. 40th Southeastern Symposium on, vol., no., pp.284-288, 16-18 March 2008 doi: 10.1109/SSST.2008.4480238
- [10] T. Neumann, "Modeling Insect Compound Eyes: Space-Variant Spherical Vision", *Lecture Notes in Computer Science*, Springer, pp. 360-367, 2002.
- [11] P. Burelli, L. Di Gaspero, A. Ermetici and R. Ranon, "Virtual Camera Composition with Particle Swarm Optimisation", *Lecture Notes in Computer Science*, pp. 130-141, Volume 5166/2008.
- [12] K. Deb, "Multi-objective optimisation using evolutionary algorithms", Wiley, 2002
- [13] Q. Zhang, H. Li, "MOEA/D: A Multi-objective Evolutionary Algorithm Based on Decomposition", *Evolutionary Computation*, *IEEE Transactions on*. vol.11 issue 6, pp. 712-731. 2008
- [14] R. Roy, J. Mehnen, *Technology Transfer*. In *Studies in Computational Intelligence Vol 88: Evolutionary Computation in Practice*, Eds. Kacprzyk, T. *et al.*, Springer, pp. 263-281, 2008.
- [15] T.P. Breckon, S.E. Barnes, M.L. Eichner, and K. Wahren, "Autonomous Real-time Vehicle Detection from a Medium-Level UAV," *Proc. 24th International Unmanned Air Vehicle Systems*, 29.1-29.9 (2009).

- [16] J. Sokalski, T. P. Breckon, I. Cowling, "AUTOMATIC SALIENT OBJECT DETECTION IN UAV IMAGERY". 25th Bristol International UAV Systems Conference Springer pp. 11.1-11.12.
- [17] P. Ostiak, Implementation of HDR panorama stitching algorithm. In Proceedings of the 10<sup>th</sup> CESC Conference, Castá-Papiernicka Centre, Slovakia, April 24-26, 2006.
- [18] R.Fernald, Casting a genetic light on the evolution of eyes. Science 313 (5795): 1914-1918. vol. 313 no. 5795 pp. 1914-1918 2006.
- [19] [http://www.codeproject.com/KB/audio-video/Motion\\_Detection.aspx](http://www.codeproject.com/KB/audio-video/Motion_Detection.aspx)
- [20] <http://news.bbc.co.uk/1/hi/world/americas/4808342.stm>
- [21] T Brown, [www.infinite.world.org](http://www.infinite.world.org), "Understanding autonomous system design through collective intelligence communication, task allocation and self-organisation in New World army ant swarms" February, 2003.
- [22] Pacheco, P. "Parallel Programming with MPI", Morgan Kaufmann, 1996.
- [23] Snir, M. "MPI: Core v. 1: The Complete Reference", MIT Press, 1998.
- [24] McMenemy K. "A Hitchhiker's Guide to Virtual Reality", A. K. Peters Publisher, 2007.
- [25] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman, "Geodesic Star Convexity for Interactive Image Segmentation" in CVPR, June 2010.
- [26] M. Ying, J. Jingjue, B. Fulin, "3D-City model supporting for CCTV monitoring system", Symposium on Geospatial Theory, Processing and Applications, Ottawa, 2002.
- [27] G. Heredia , F. Caballero, I. Maza *et al*, "Multi-UAV Cooperative Fault Detection Employing Vision-Based Relative Position Estimation", 17th World Congress, The International Federation of Automatic Control, Seoul, pp. 12093-12098. July 2008.
- [28] H. Xu, H. Guan, A. Liang, X. Yan, "A Multi-Robot Pattern Formation Algorithm Based on Distributed Swarm Intelligence", Second International Conference on Computer Engineering and Applications, 2010.
- [29] M. Brown, D.G. Lowe, "Automatic Panoramic Image Stitching using Invariant Features", International Journal of Computer Vision vol. 74 Issue 1, Vancouver, pp.59-73, August 2007.
- [30] R. W. Beard, J. Lawton, and F. Y. Hadaegh. "A coordination architecture for spacecraft formation control". IEEE Transactions on Control Systems Technology, 9(6):777{790, November 2001.
- [31] P.K.C. Wang, "Navigation strategies for multiple autonomous mobile robots moving in formation", Journal of Robotic Systems, vol. 8, no. 2, pp 177-195, 1991.

- [32] C. Reynolds, "Flocks, herds and schools: A distributive behavioural model", *Computer Graphics*, vol. 21 issue 4 pp. 25-34, 1987.
- [33] M. J. Mataric, "Designing Emergent Behaviors: From Local Interactions to Collective Intelligence", in *Proceedings, From Animals to Animats 2, Second International Conference on Simulation of Adaptive Behavior (SAB-92)*, MIT Press, 432-441.
- [34] T. Balch and R. Arkin. "Behavior-based formation control for multi-robot teams". *IEEE Transactions on Robotics and Automation*, vol. 14 issue 6 pp.926–939, 1998.
- [35] T. Balch, and M. Hybinette, "Social potentials for scalable multi-robot formations", *IEEE International Conference on Robotics and Automation*, vol. 01 pp.73-80. San Francisco, 2000.
- [36] J. Fredslund and M. J. Mataric, "Robot Formations Using Only Local Sensing and Control, Proceedings", *IEEE International Symposium on computational Intelligence for Robotics and Automation (CIRA-2001)*, Banff, Canada, Jul 29-Aug 1 2001.
- [37] M. A. Lewis and K. H. Tan, "High precision formation control of mobile robots using virtual structures", *Autonomous Robots*, vol. 4, pp. 387- 403, 1997.
- [38] B. J. Young, R. W. Beard, and J. M. Kesley, "A Control Scheme for Improving Multiple-Vehicle Formation Maneuvers", *ACC01-IEEE 1575* 2001.
- [39] E. A. Olsen, C. W. Park, J. P. How, "3D Formation Flight Using Differential Carrier-phase GPS Sensors", *Institute of Navigation GPS Meeting*, Nashville, TN, September 18, 1998.
- [40] D. Shreiner, M. Woo, J. Neider, and T. Davis, "OpenGL(R) Programming Guide: The Official Guide to Learning OpenGL(R)", Version 2 (5th Edition). Addison-Wesley Professional, August 2005
- [41] MD2 file format "<http://tfc.duke.free.fr/coding/md2-specs-en.html>"
- [42] J. L.Hansen and B. R. Cobleigh, "Induced Moment Effects of Formation Flight Using Two F/A18 Aircraft", Technical Report NASA/TM-2002-210732, NASA Dryden Flight Research Centre, Edwards, California. [http://www.nasa.gov/centers/dryden/pdf/88744main\\_H-2502.pdf](http://www.nasa.gov/centers/dryden/pdf/88744main_H-2502.pdf)
- [43] P. Binetti , K.B. Ariyur, F. Bernelli and M. Krstic "Formation Flight Optimisation Using Extremum Seeking Feedback" *Journal of Guidance Control and Dynamics* vol 6 No 1 January-February 2003
- [44] A. Gopalarathnam, "Aerodynamic Benefit of Aircraft Formation Flight. *Encyclopaedia of Aerospace Engineering*" 2010
- [45] F.Rafi, S.M.Khan, K Shafiq and M.Shah, "Autonomous Target Following by Unmanned Aerial Vehicles", *SPIE Defence and Security Symposium 2006*, Orlando FL.

[46] [http://scienceblogs.com/grrlscientist/2009/07/30000\\_facets\\_give\\_dragonflies.php](http://scienceblogs.com/grrlscientist/2009/07/30000_facets_give_dragonflies.php)

[47] W. Li and W. Shen. "Swarm behavior control of mobile multi-robots with wireless sensor networks". J. Netw. Comput. Appl. 34, 4 (July 2011), 1398-1407. DOI=10.1016/j.jnca.2011.03.023 <http://dx.doi.org/10.1016/j.jnca.2011.03.023>

[48] K. S. Pratt, R. Murphy, S. Stover, and C. Griffin. "CONOPS and autonomy recommendations for VTOL small unmanned aerial system based on Hurricane Katrina operations". J. Field Robot. 26, 8 (August 2009), 636-650. DOI=10.1002/rob.v26:8 <http://dx.doi.org/10.1002/rob.v26:8>

[49] A. Ryan, M. Zennaro, A. Howell, R. Sengupta and J. K. Hedrick, "An Overview of Emerging Results in Cooperative UAV Control", Proceedings of the 43rd IEEE Conference on Decision and Control, vol.1 pp.602-607, December 2004

## 8 Glossary of Terms and Acronyms

---

Acronym	Description
MPICH2	Message Passing Interface (MPI-2 standard)
MPI	Message Passing Interface
OpenGL	Open Graphics Library
GLUT	OpenGL Utility Toolkit
MFC	Microsoft Foundation Classes (library)
GUI	Graphical User Interface
LM	Lockheed Martin
WP x	Work Package x
Tx	Task x
Dx	Deliverable x
UAV	Unmanned Aerial Vehicle
LMC	Lockheed Martin Corporation
PSO	Particle Swarm Optimisation
HDR	High Dynamic Range
OO	Object Oriented
CCTV	Closed Circuit Tele-Vision
FoV	Field of View
GPS	Global Positioning System
DGPS	Differential Global Positioning System
VFI	Visual Factor Index
SN	Sensor Network
MMR	Multiple Mobile Robots

## 9 Appendix

---

### Images for Example 1



Figure 41



Figure 42



Figure 43



Figure 44





Figure 45



Figure 46

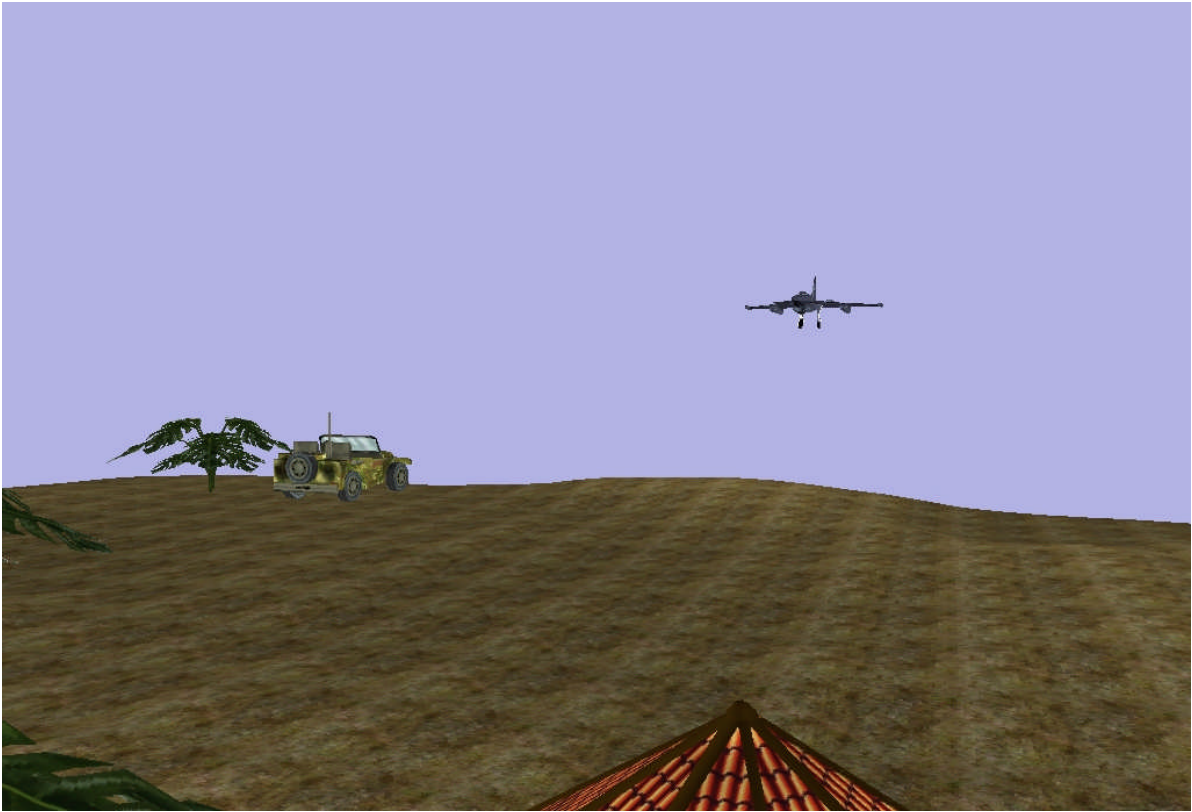


Figure 47



Figure 48



Figure 49



Figure 50



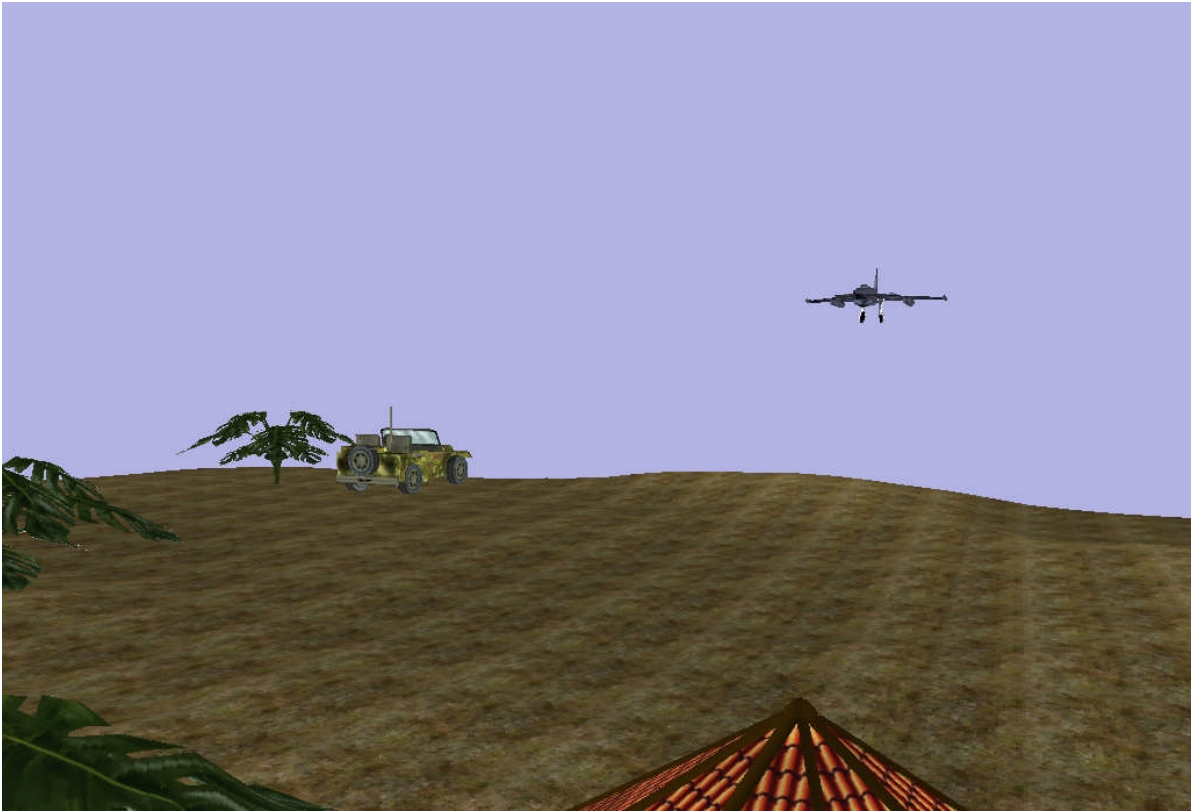


Figure 51



Figure 52

## Images for Example 2



Figure 53



Figure 54





Figure 55



Figure 56





Figure 57

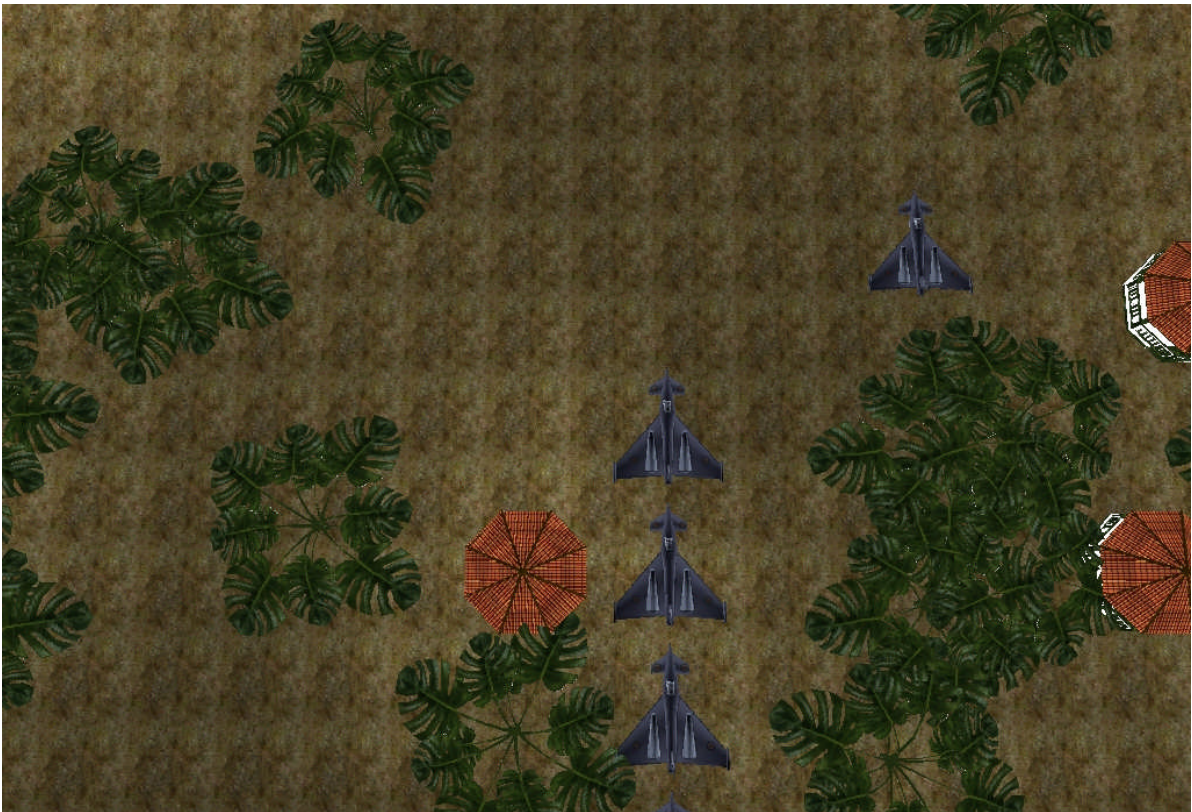


Figure 58





Figure 59



Figure 60





Figure 61



Figure 62





Figure 63



Figure 64



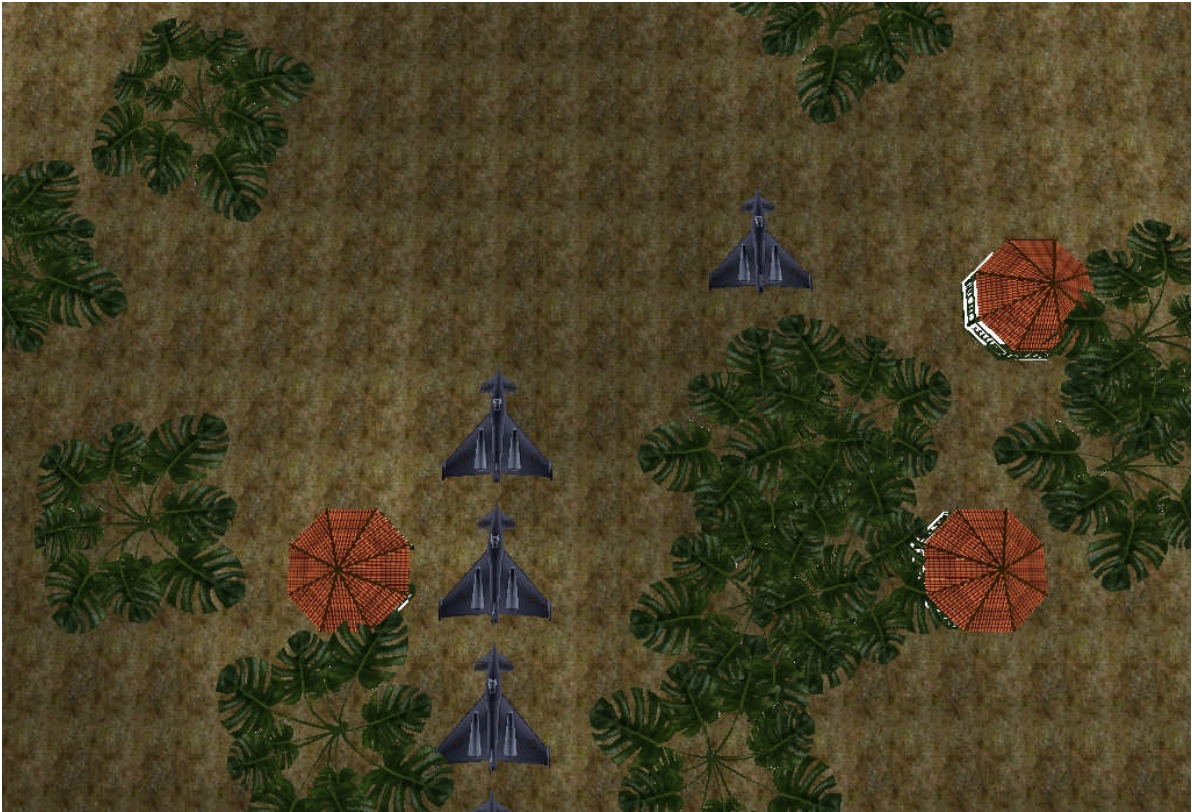


Figure 65



Figure 66



### Images for Example 3

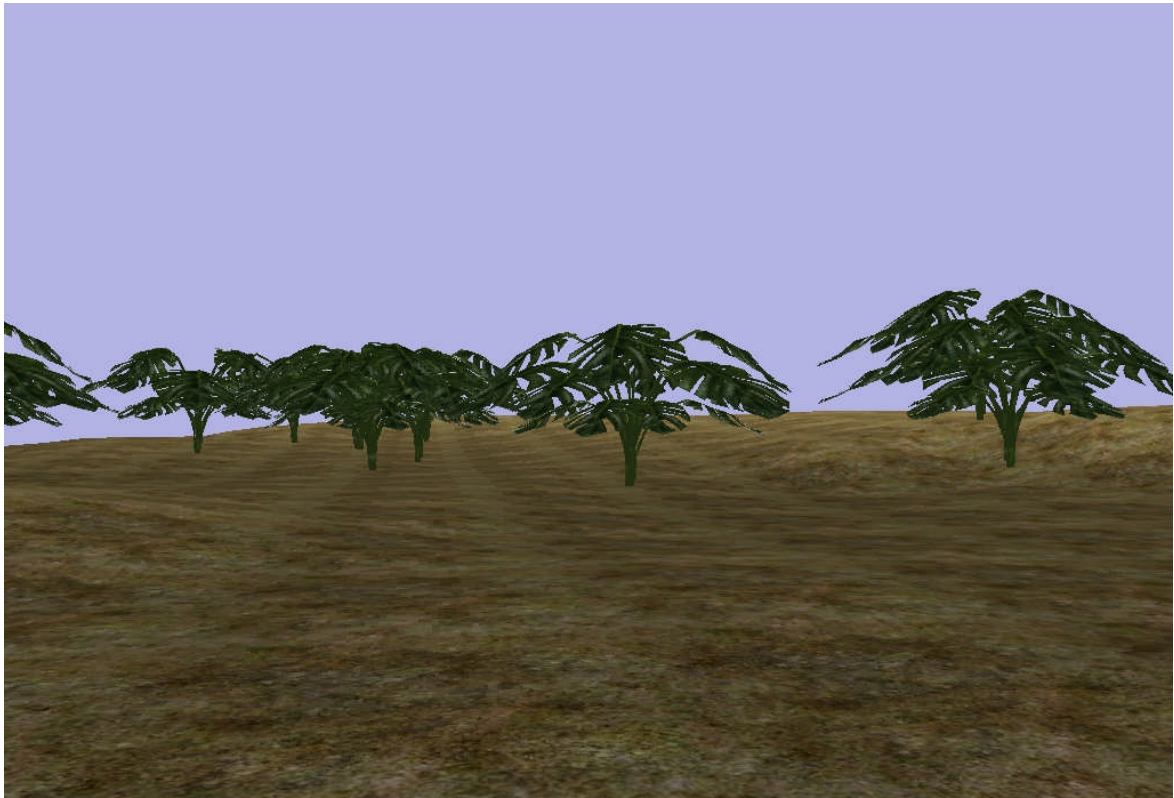


Figure 67

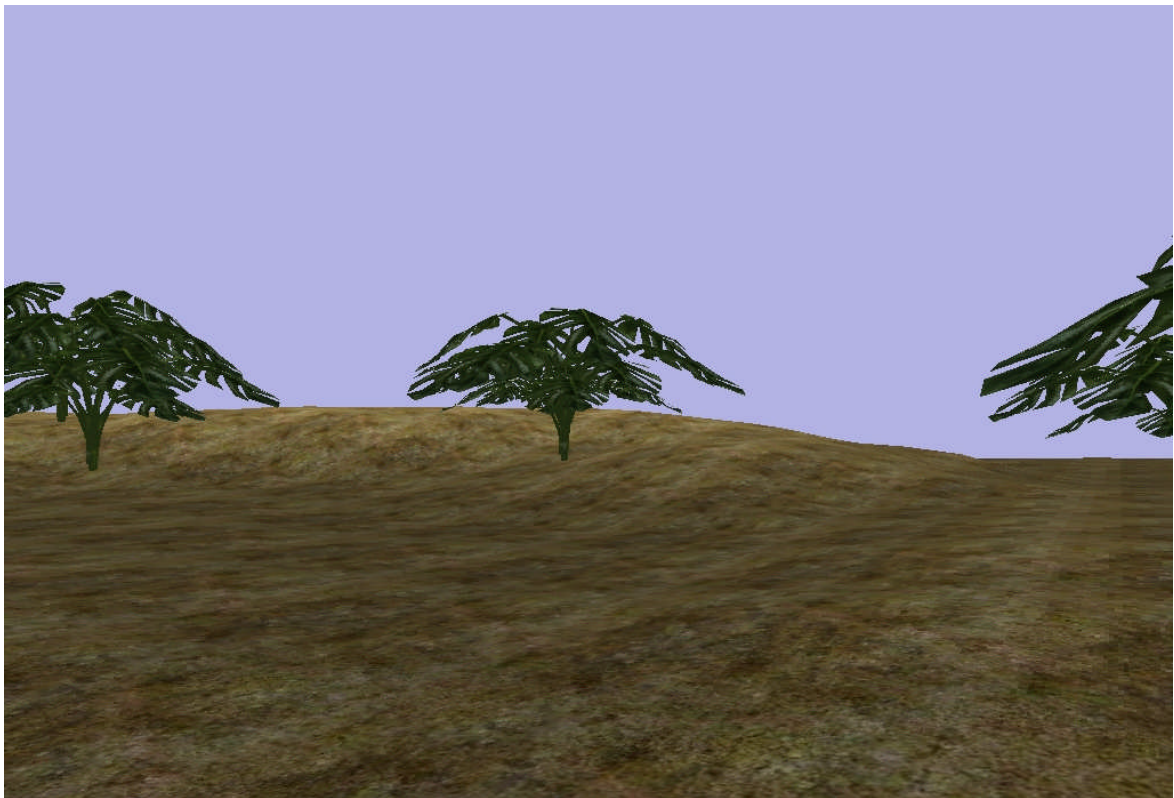


Figure 68

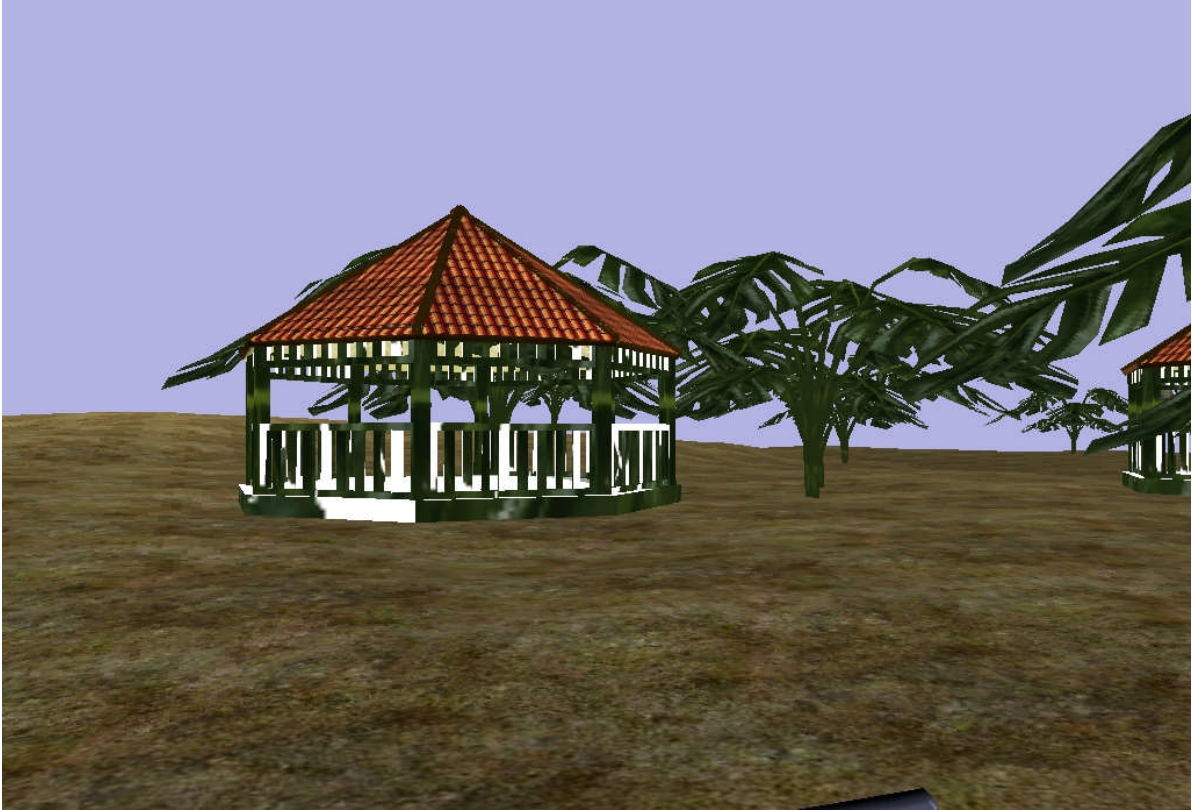


Figure 69

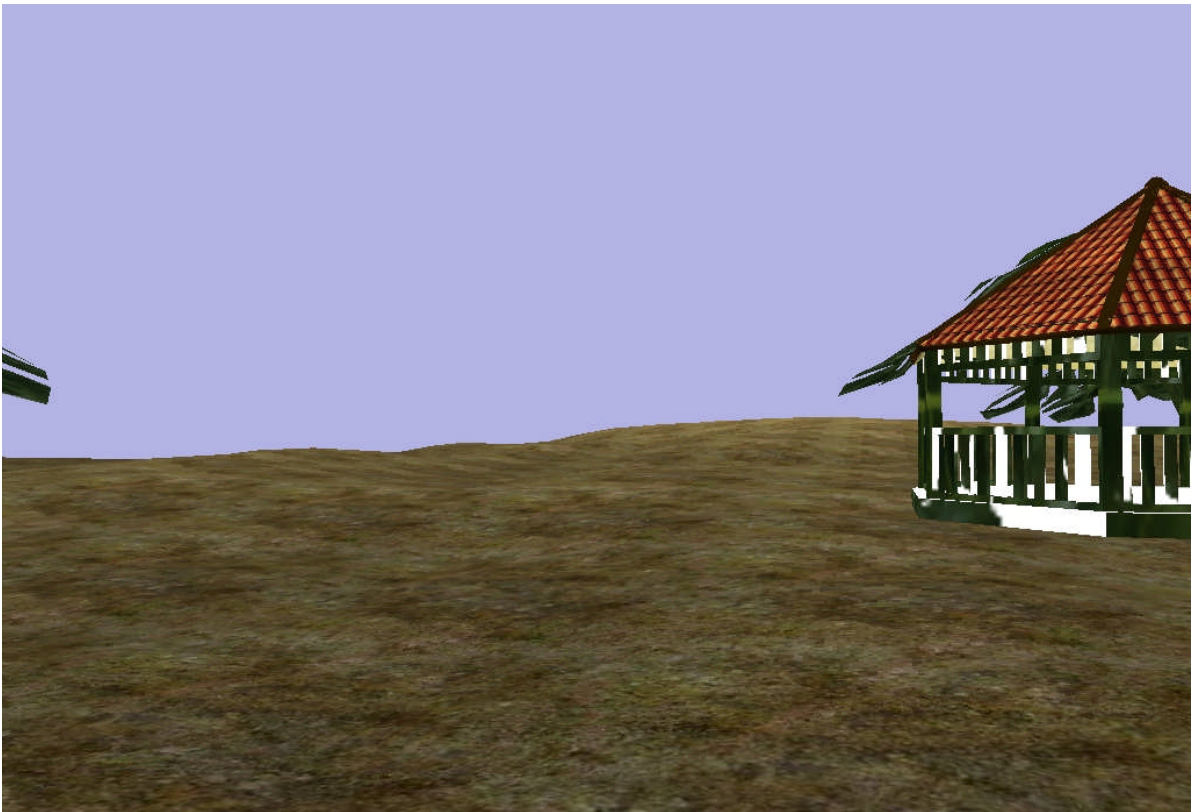


Figure 70





Figure 71



Figure 72



Figure 73



Figure 74



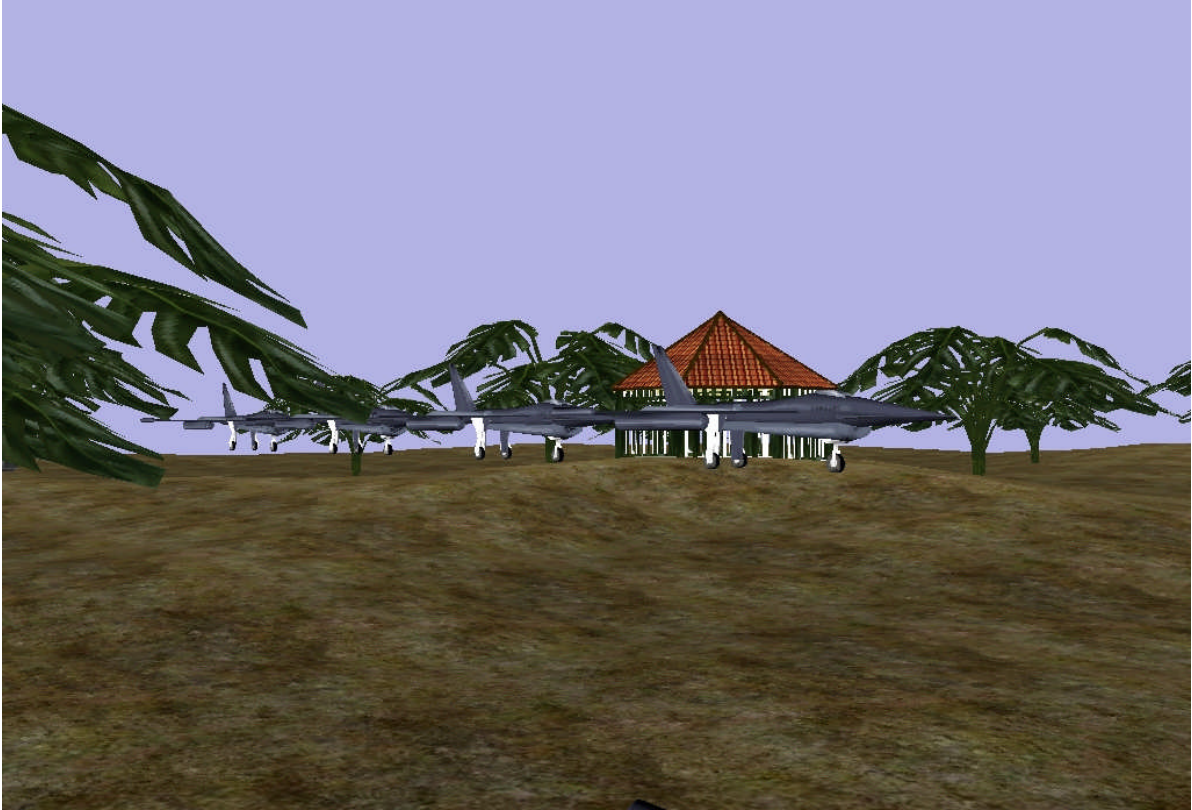


Figure 75

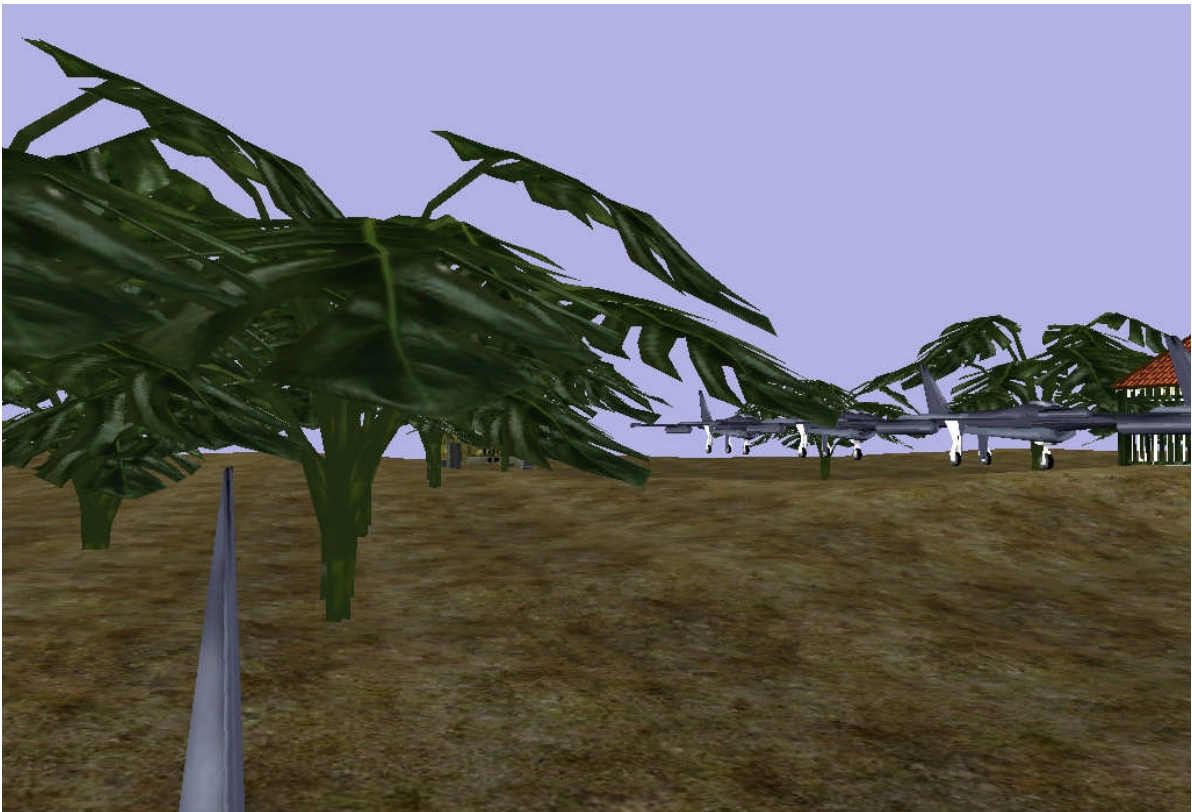


Figure 76



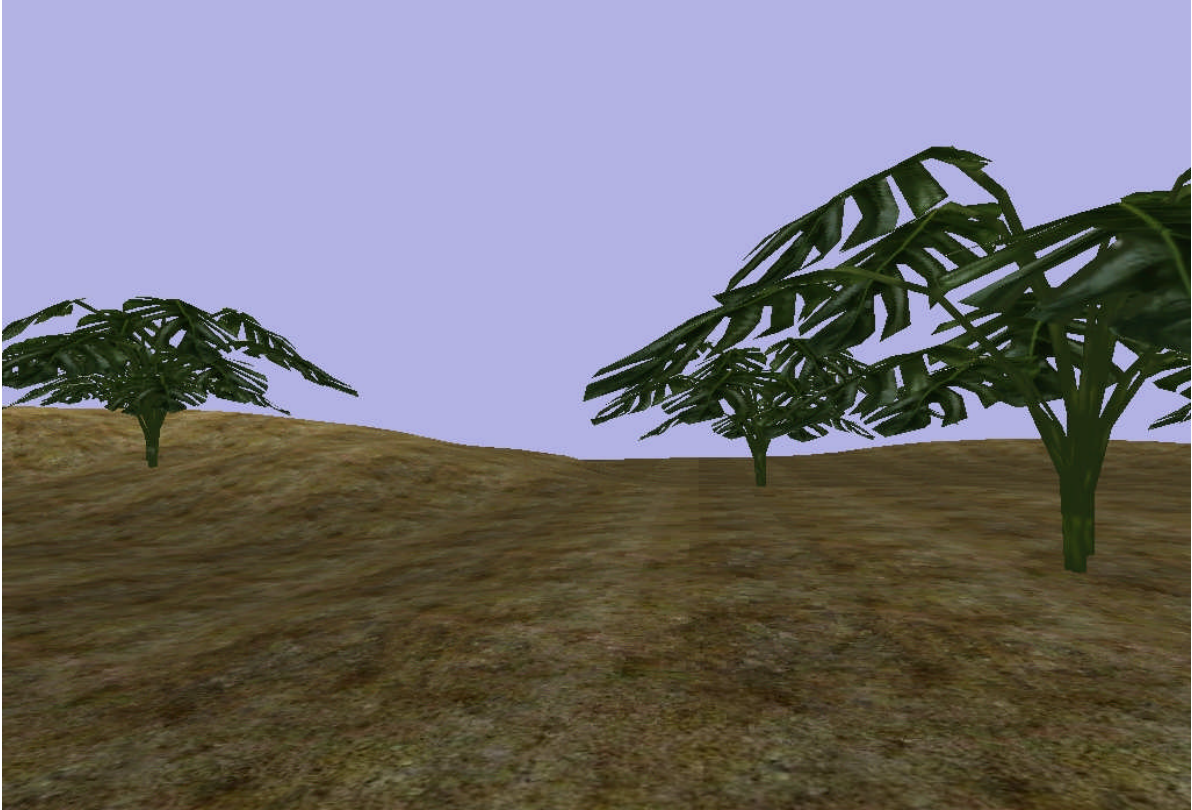


Figure 77



Figure 78

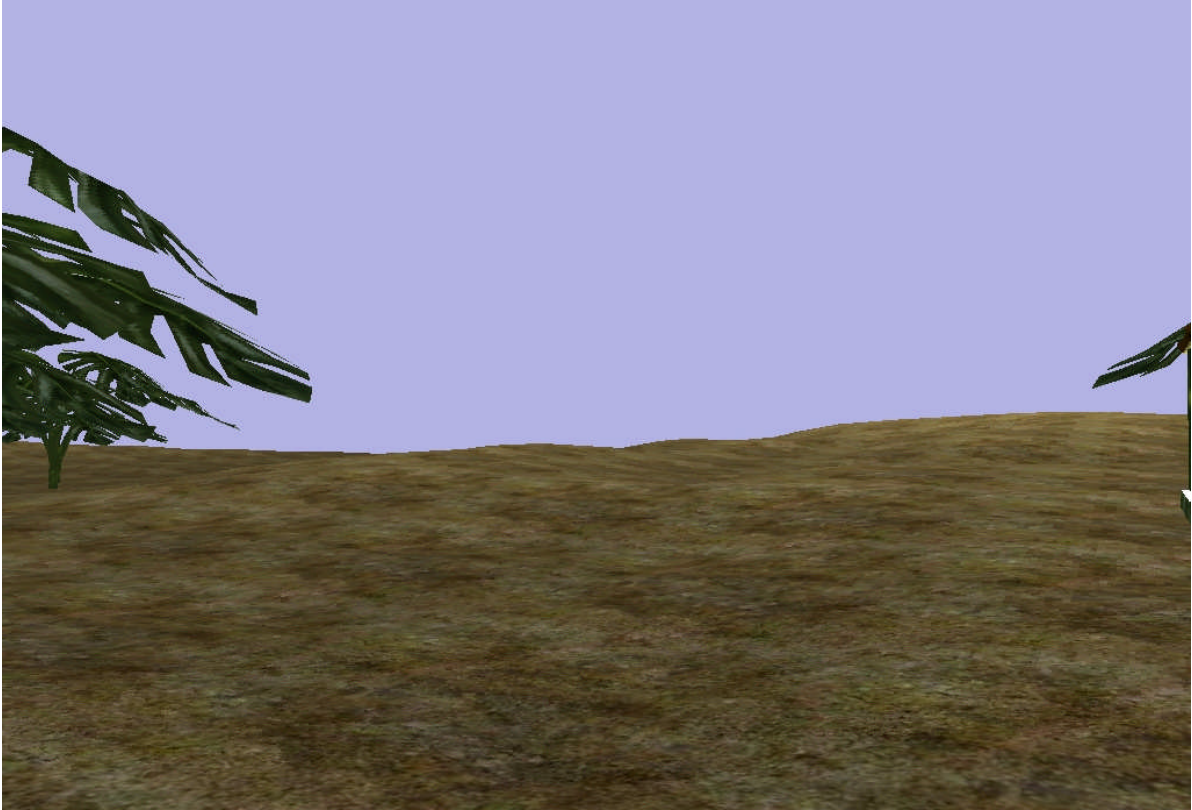


Figure 79



Figure 80



## Images for Example 4



Figure 81



Figure 82





Figure 83



Figure 84





Figure 85



Figure 86





Figure 87



Figure 88





Figure 89



Figure 90





Figure 91



Figure 92





Figure 93



Figure 94



## Images for Example 5

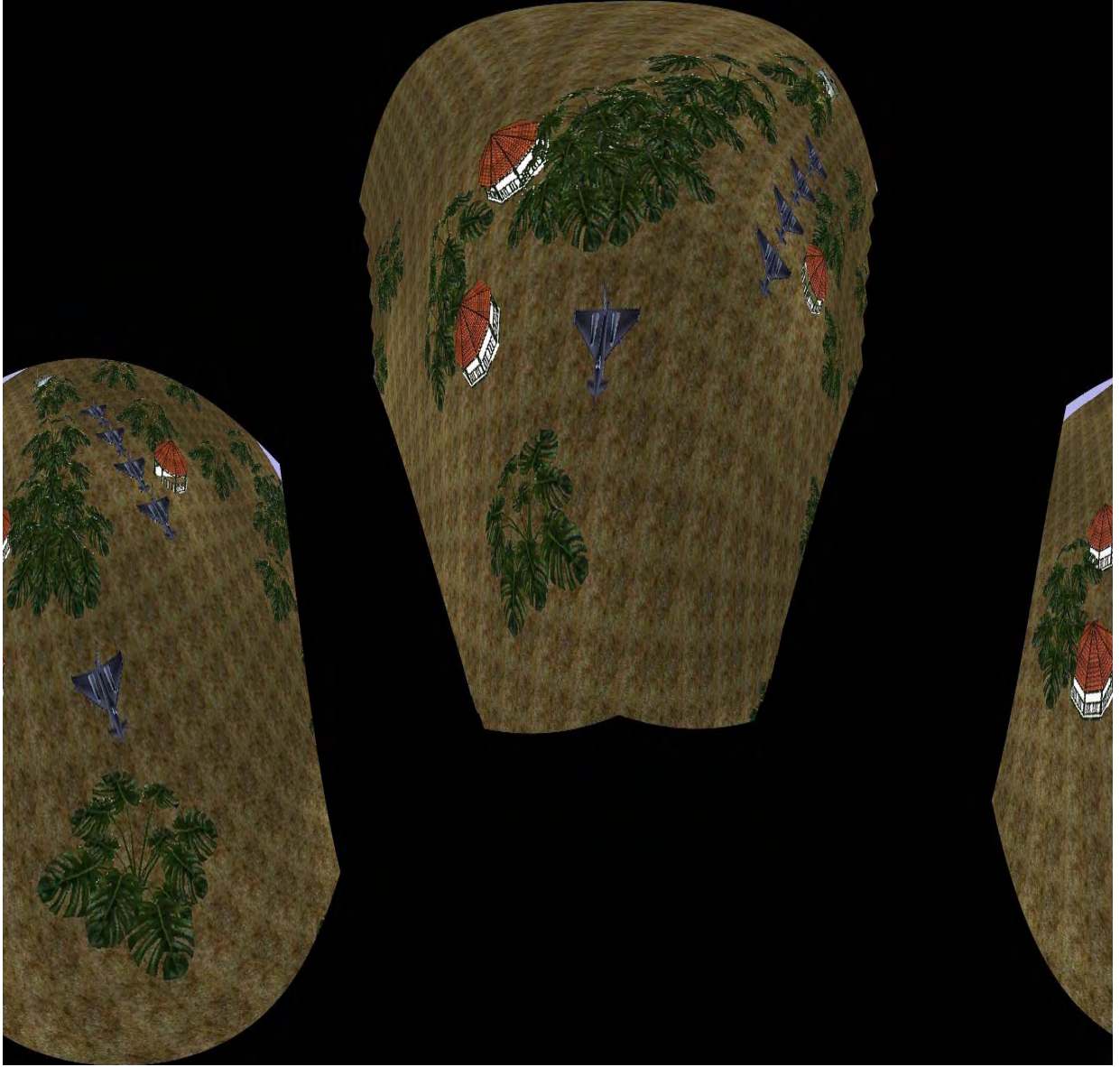


Figure 95 Panorama Generated by using Figures 96-105

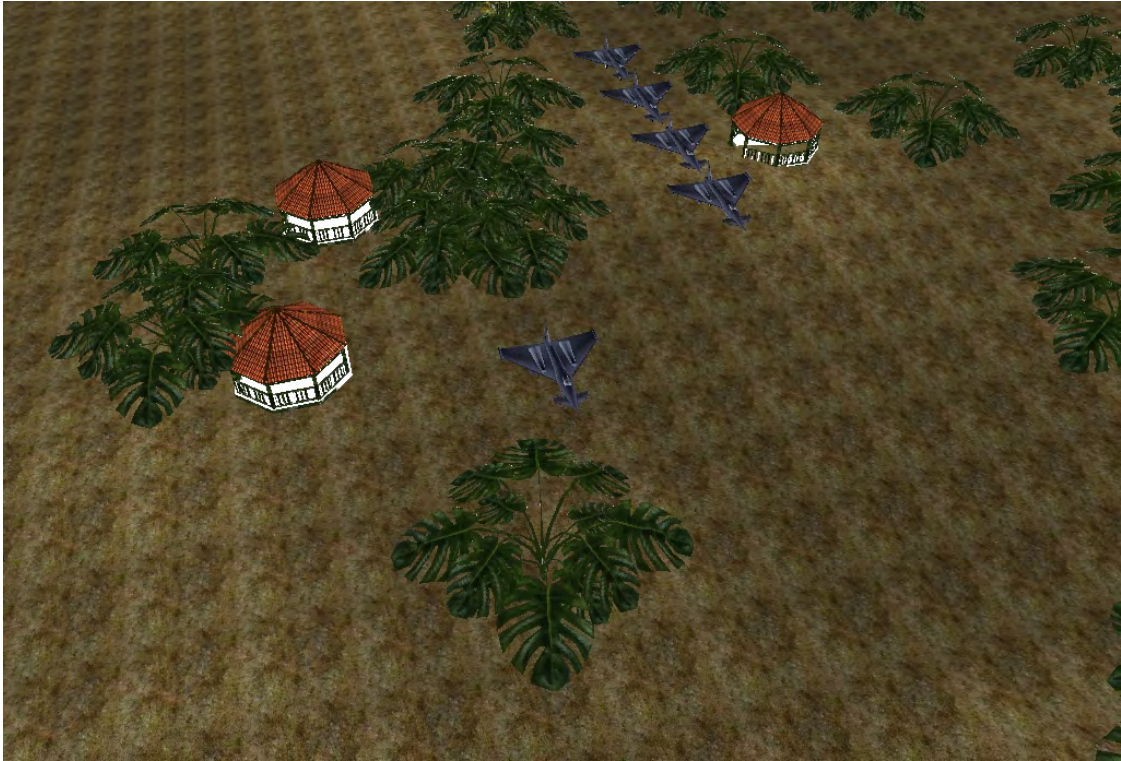


Figure 96

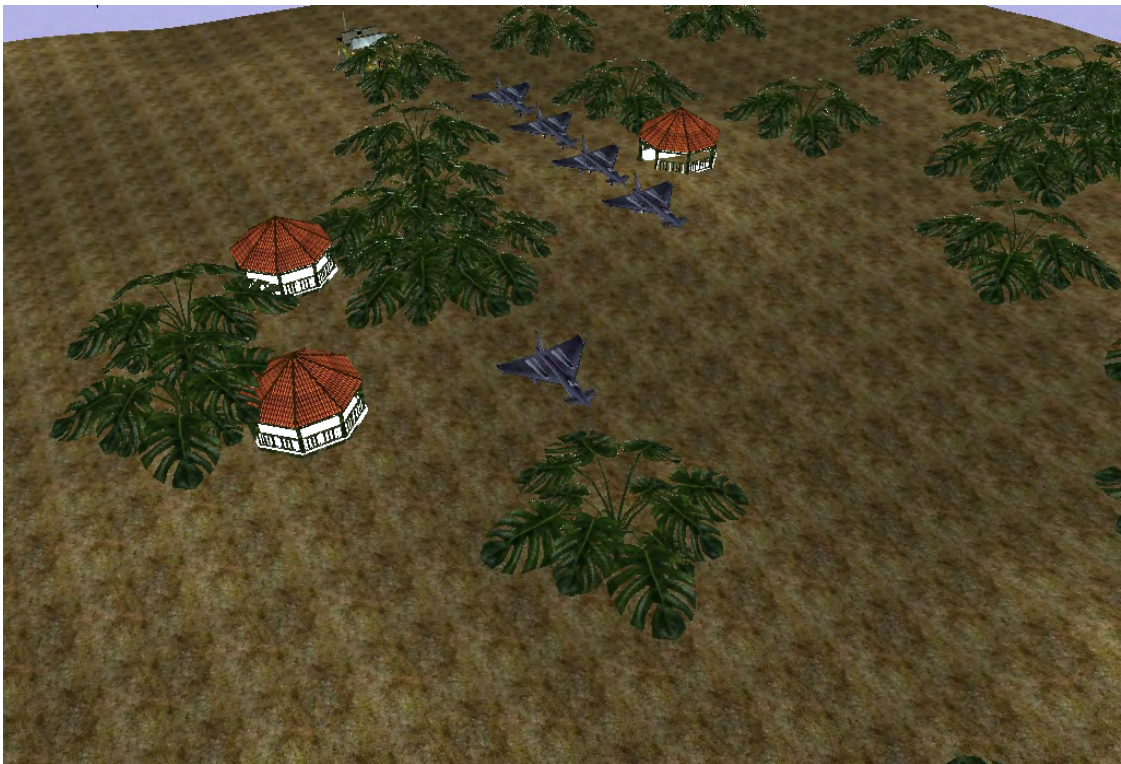


Figure 97





Figure 98

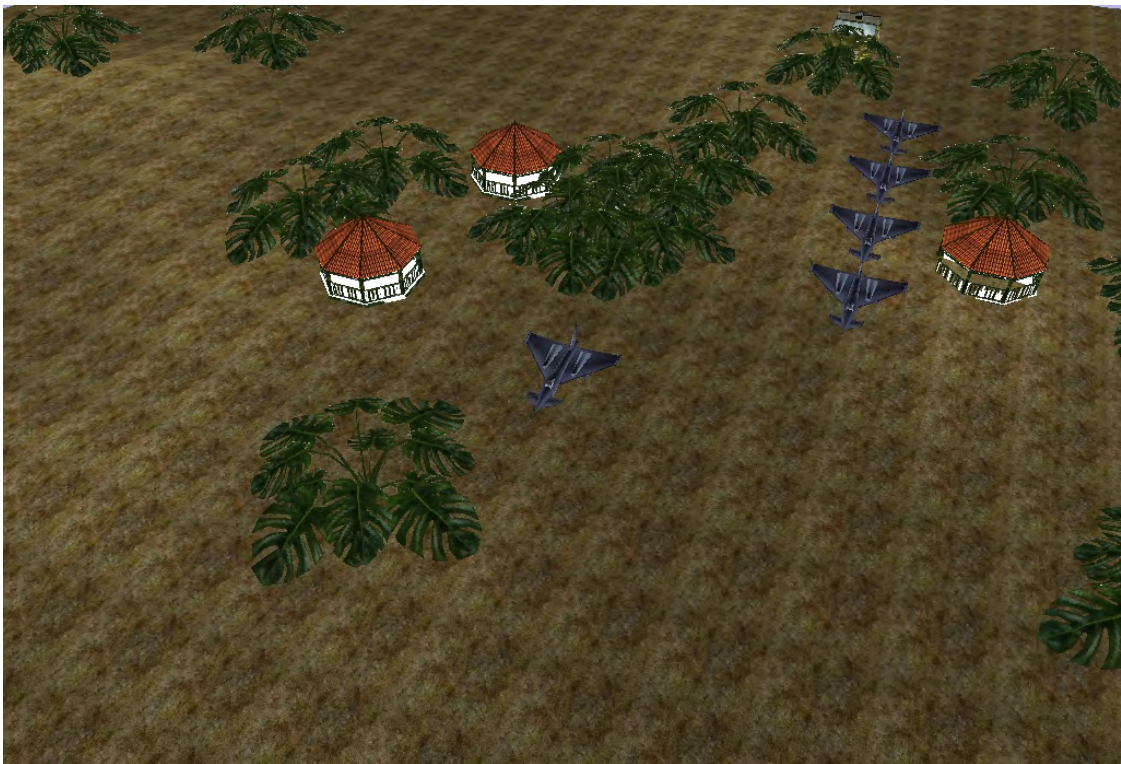


Figure 99





Figure 100

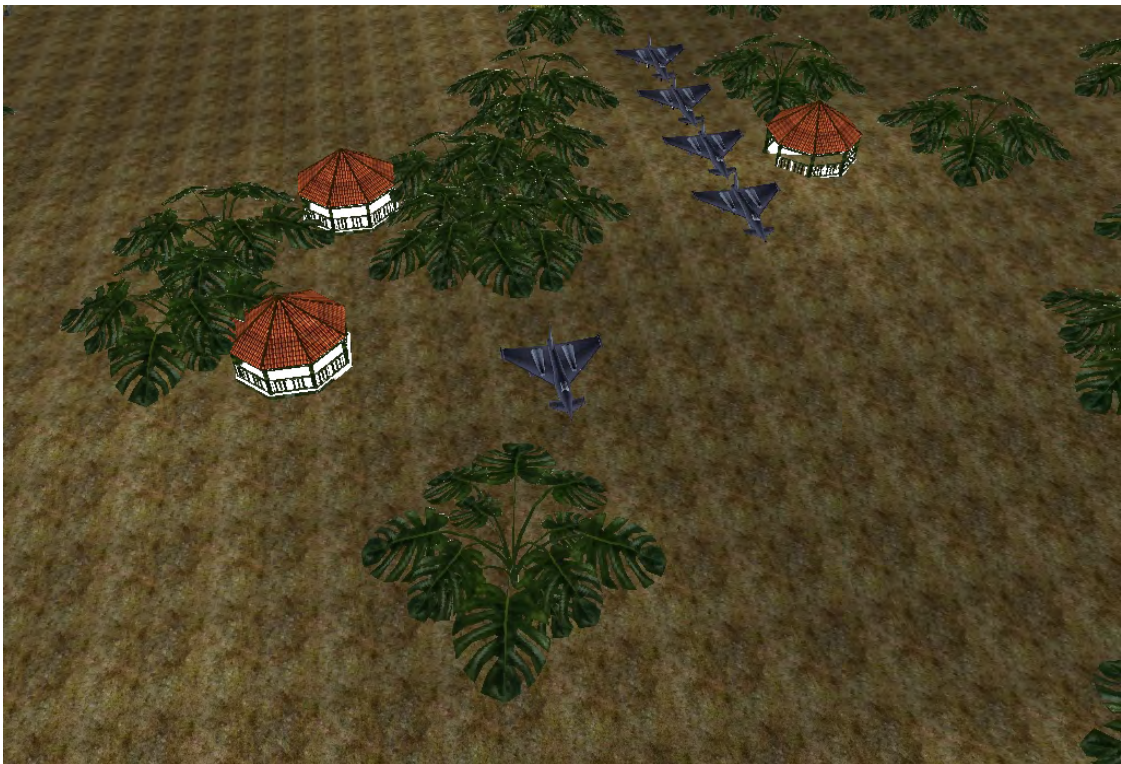


Figure 101





Figure 102



Figure 103





Figure 104

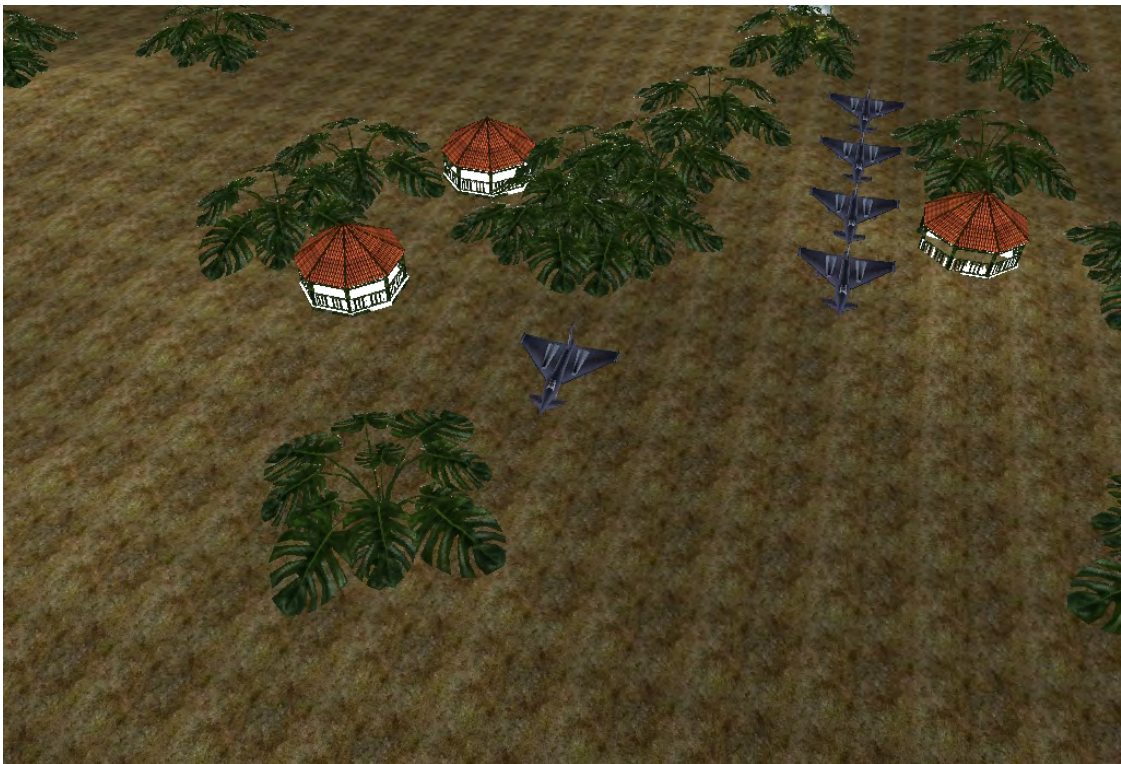


Figure 105