

Design and Implementation of Autonomous Fish Behavior in Virtual Interactive Aquarium Application

Allin Junikhah

Asia Malang Institute of Technology and Business, Indonesia
Corresponding author. Email: non.junic@gmail.com

ABSTRACT

The Interactive Aquarium Application is an application that represents the real environment of the aquarium into an interactive virtual environment. Children as users can interact with objects in the aquarium through user gesture processing. Several supporting elements are needed in developing the Interactive Aquarium application, such as a visual representation of the aquarium environment through Non-Playable Character (NPC) objects or in simulations called autonomous characters. One of the autonomous characters in this game is fish. Several techniques are needed to build artificial intelligence in fish, for example, the behavior of the control movements (fish steering behavior) to simulate fish behavior.

Keywords: *Interactive Aquarium, simulation, autonomous character, Finite State Machine, steering behavior.*

1. INTRODUCTION

Steering behavior is part of motion behavior. Research on this topic continues to be developed in the field of computer graphics. Steering behavior by Craig W Reynold is widely used for various fields, including robotics, artificial intelligence, and artificial life. In addition, steering behavior is also widely used in the entertainment genre, such as games and animation [12].

This study aims to design and implement a behavioral model of an autonomous fish agent (autonomous fish) as an interactive object in an interactive aquarium application.

Activities of autonomous fish behaviors that are implemented include fish behavior in general, such as the steering behavior on the fish like wandering, schooling, seeking, fleeing, the ability to detect nearby food, the ability seen within a certain distance (sight radius), and also implementation of energy as a parameter of fish life. Autonomous fish behavior is modeled by the concept of a Rule-Based System represented in the form of a Finite State Machine. Implementation of the model used in the fish-shaped autonomous 2D visual.

2. THEORETICAL

2.1. Autonomous Character

An autonomous character is a type of autonomous agent intended for use in computer animation and

interactive media such as games, simulations, and virtual reality. These agents represent characters in a story or game and have the ability to improvise actions. This is in contrast to a character in an animated film whose actions are written in advance, and an avatar in a game or virtual reality, the action of which is directed in real-time by the player. In games, autonomous characters are usually called Non-Playable Character [8].

2.2. Finite State Machine (FSM)

FSM is a tool or model with a finite number of states at a time and can operate input to make a transition from one state to another or cause an output to occur [2]. FSM can only be in one state at a time and is a method that can display character behavior in a simulation or game. Transitions in FSM are obtained from the environment or system and then processed according to the internal data on the system. The main idea of FSM is to decompose a behavior into several states, making it easier for developers at the early design stage [4].

2.3. Steering Behaviour

Steering behavior is a collection of behavior from a unit to produce a steering force that will affect the movement and direction of the unit. In steering behavior, a vehicle model is needed, the coordinates are addressed, and the direction or heading is set at that time. It can also

be obtained a new direction for the vehicle. An automated moving agent consists of three important parts [9]:

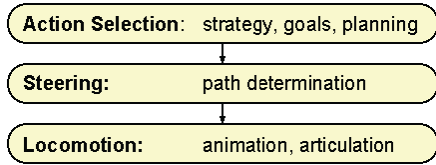


Figure 1 Motion path behavior schema [10]

3. DESIGN AND IMPLEMENTATION

Interactions that can be carried out by users or players in interactive aquarium applications, especially those related to autonomous fish agents, include:

- a. **Feeding fish:** Feeding fish can be done by displaying certain gestures. Gesture recognition is used as a trigger to generate worm objects as fish food. Fish with a visual perception that can reach the worm object will approach and catch the food.
- b. **Chasing fish:** One of the fish behaviors in the Interactive Aquarium application is to avoid the position of the user's hand. Users can play with chasing fish and find out the response or interactivity of the autonomous fish when the user puts his hand on a group of fish.

3.1. System Requirement Analysis

Some of the supporting elements needed in the Interactive Aquarium application, especially fish objects, are as follows:

- Fish character model to use
- Type of fish behavior to be implemented
- Behavioral attributes in fish such as energy and swimming speed
- Supporting technology to be used in developing fish behavior
- Multiple artificial intelligence implemented on fish
- Specifications for implementation requirements
- Minimum specifications for testing devices

3.2. Autonomous Fish Character Model

In developing fish behavior models in the Interactive Aquarium application, a two-dimensional (2D) visual character model is used. Cartoon character models are used as realistic fish representations but still feature coloring elements that are more fun for children.



Figure 2 Two-Dimensional (2D) character model

3.3. Types of Fish Behaviour

The types of fish behavior that will be implemented in the Interactive Aquarium application are general fish behavior, including the following:

- NPCs have the ability to wander
- NPCs have the ability to catch food
- NPCs have the ability to avoid users (hands)
- Some NPCs have the ability to group according to their type
- NPCs have the ability to detect nearby food
- NPCs have the ability to see within a certain distance
- Energy implementation on NPC
- The system's ability to simulate NPCs dying when not meeting certain energy

3.4. Attributes Supporting Fish Behaviour

Some of the fish behavior implemented in the application has several supporting attributes such as swimming speed and energy. Swimming speed is used as a representation of the movement of fish objects from one point to another. In comparison, the energy in fish represents the strength of the fish or as a parameter of fish life.

Table 1

Details of fish behavior supporting attributes

No	Activity Description	Energy	Speed
1.	Wander	Reduced	Medium
2.	Seek	Reduced	Maximum
3.	Flee	Reduced	Maximum
4.	Schooling	Reduced	Medium
5.	Eat food	Increase	-
6.	Exposed to pollution	Reduced	Medium
7.	Dead fish	zero	Medium (float)

3.5. Supporting Technologies

An autonomous fish system in the virtual interactive aquarium application requires some technological support in its development. The following are some of the supporting technologies used.

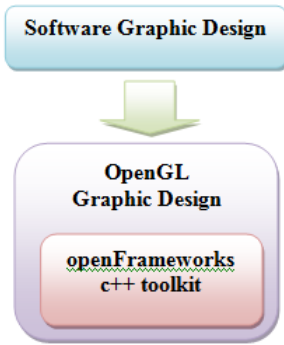


Figure 3 Supporting technology

a. *Graphic Design Software*

Graphic Design Software is used to design and edit fish objects used in interactive Aquarium applications.

b. *OpenGL Graphic Design*

Graphic Application Programming Interface is needed as a graphics library supporting graphic programming for developing interactive aquarium applications. The graphic API used in the development of this application is OpenGL.

c. *C++ Toolkit*

OpenFrameworks c++ toolkit is a set of libraries consisting of several classes written in the c++ programming language to assist in facilitating the efficiency of writing program code in the development of applications that require graphics and audio support.

3.6. *Implementation Requirements Specification Tool*

The following are the device specifications required in the implementation:

a. *Software Specification*

- Operating System: Windows 7 Ultimate Service Pack 1
- C + + IDE: 10:05 Codeblock version
- Library: OpenFrameworks
- Graphic Editor: Adobe Photoshop

b. *Hardware Specifications*

- Processor: SU4100 (2MB cache, 1:30 GHz, 800 MHz FSB)
- Memory: 2 GB DDR2
- VGA: Mobile Intel® Graphics Media Accelerator 4500MHD

3.7. *Minimum Specifications Testing Devices*

The following are the minimum specifications of the required test devices:

a. *Software Specification*

- Operating System: Windows XP Professional SP 1
- Drivers DirectX version 9.0 or higher
- Driver sound card

- VGA driver

b. *Hardware Specifications*

- Processor : Pentium 4 1,2 GHz
- Memory: 2 GB
- VGA: 64 MB
- Hard disk: 320 GB
- Multimedia: 16 bit stereo sound card or higher

3.8. *Autonomous Fish Behaviour Design*

In designing the behavior model of autonomous fish, the Finite State Machine method will be used, representing several types of behavior in fish.

Here are some states used in FSM:

- Wandering
- Catching food
- Eat
- Flee
- Schooling
- Exposed to pollution
- Dead fish

Figure 4 shows the behavior implemented on autonomous fish through state representation and several transitions as a liaison between states, which is a condition that must be met to move from one state (current state) to another state (next state).

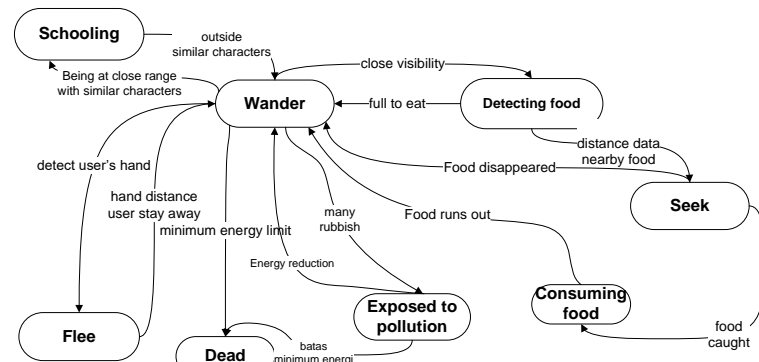


Figure 4 State diagram of autonomous fish behavior

3.9. *Implementation of Steering Behaviour*

In the Interactive Aquarium application, autonomous fish, as one of the important objects in this application, are given steering behavior. This behavior is implemented on the object as an object control in determining the direction of motion. Two types of techniques in steering behavior that are implemented include catching (seek and pursuit), avoiding (flee), wandering (wander), and grouping (schooling) techniques.

3.10. *Seek and Pursuit Steering Technique*

In the implementation of steering behavior, several techniques used, such as seek, pursuit, flee, wander, and schooling, will implement the same steering function in

the process of steering fish to a certain point. In the seek and pursuit technique, the user player determines the target as the coordinate point of the food to be caught by the fish object.

The steering process in capturing behavior is the process of driving the object to the target position. There is a change in the direction of speed in it. If the target position with the object location is not more than zero, then the target position is considered to be at the object location, and no steering process is carried out.

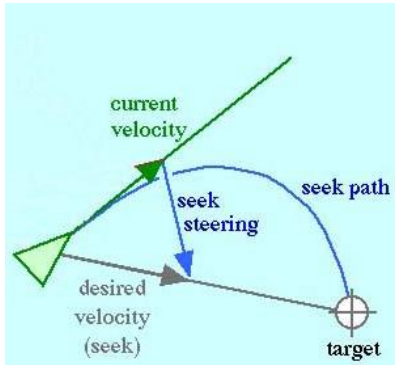


Figure 5 Steering^[10]

- (1) $velocity = velocity + acceleration$
- (2) $desired_velocity = normalize(target - location) * max_speed$
- (3) $steering = desired_velocity - velocity$



Figure 6 Seek behavior

Another technique used to catch food targets by fish is the pursuit technique. The pursuit technique can be used to steer dynamic targets. The pursuit predicts the target's location, so the movement is not repeated with the condition of the target moving.

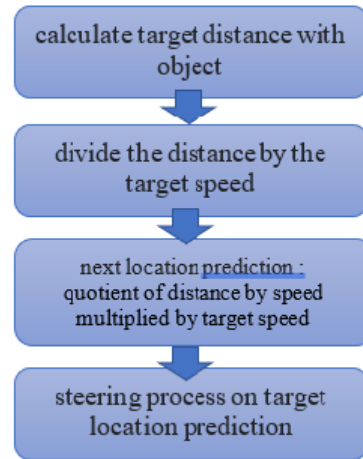


Figure 7 Pursuit behavior computing^[10]

3.11. Flee Steering Technique

In the flee behavior of the fish object, steering will be carried out to avoid certain points, such as the user's hand object. This behavior is the opposite of seek behavior.

In the flee process, a calculation is made between the position of the fish object and the object to be avoided, in this case, the user's hand. This is done to find out how far the fish is from the user's hand. Because the user's hand as the object to be avoided is dynamic, sometimes the distance of the fish is at zero or less than zero, which indicates that the fish is in the position of the radius area of the object's hand. The avoidance of fish is analogous to the fear that fish have.

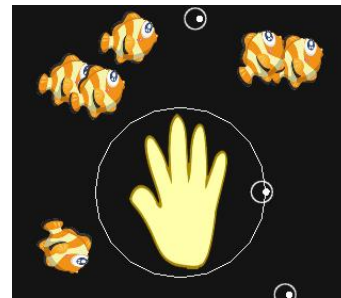


Figure 8 Flee behavior

3.12. Wander Steering Technique

One of the fish activities that can be observed is swimming without direction with the aim of relaxation. This activity is also called wander.

The purpose of wander is first to determine the turn angle obtained from random results with a certain range and then to steer the fish object to the target point resulting from the process of summing the wander circle locations with offsets.

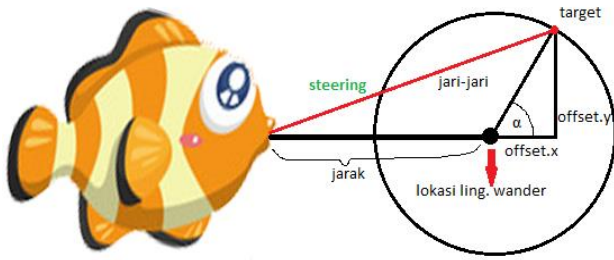


Figure 9 Variabel wander

$$Offset.x = radius \times \cos(\alpha) \quad (4)$$

$$Offset.y = radius \times \sin(\alpha) \quad (5)$$

$$Target = Wander\ circle\ location + Offset \quad (6)$$

3.13. Schooling Steering Method

Schooling is a behavior in which the fish character object performs group activities when encountering the same species in a certain environment. In group behavior, autonomous fish have three sub-behaviors that form schooling behavior.



Figure 10 Schooling behavior

a. Cohesion

Finds the average position of the neighbors in the neighborhood and performs a displacement on that position.

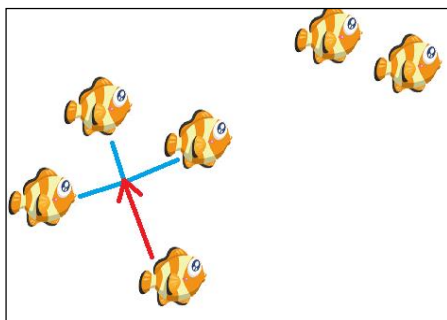


Figure 11 Cohesion

b. Alignment

Finds the average direction on all neighbors and performs a displacement in that direction. This is done by calculating the average speed on all neighbors.

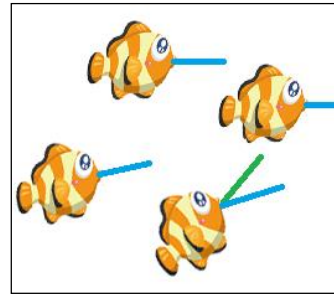


Figure 12 Alignment

c. Separation

The object will always maintain a certain distance between its position and the position of its neighbors in its environment.

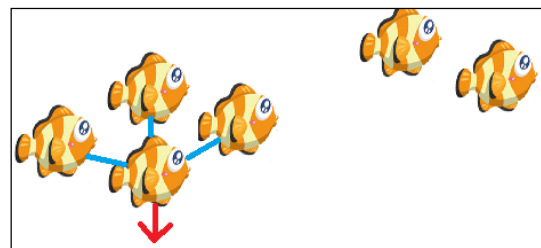


Figure 13 Separation

3.14. Nearby Food Search Technique

In catching fish food, it is given the ability to catch food closest to the fish's position. If several foods are fed to fish at the same time with different food coordinate positions, the fish can identify the closest food from their position.

In this capability, the Euclidean Distance algorithm is used to determine the distance between the fish and the position of the food.

$$Fish_foodN_distance = \sqrt{(foodNlocation - fishlocation)^2} \quad (7)$$

3.15. Autonomous Fish Visibility (Sight Radius)

Autonomous fish also have certain visibility (sight radius) in determining food catching. In the Interactive Aquarium application, the sight radius is described in the form of a circle with the position of the fish object as the axis.

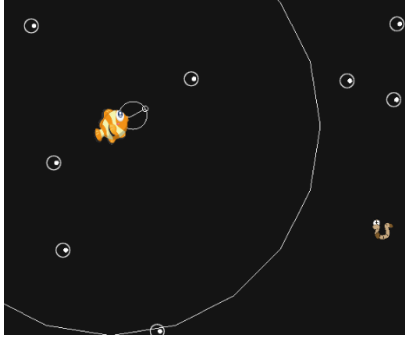


Figure 14 Food outside the sight radius

3.16. The attribute of Autonomous Fish

In its implementation, this virtual fish object is given several attributes to support artificial intelligence capabilities. These attributes include speed and energy, as mentioned in the design. The energy possessed by the fish is used as a parameter of the condition of the fish. Fish energy will change along with the activities carried out by the fish object.

Table 2

Details of fish behavior supporting attributes

No	Activity Description	Energy Value
1.	Wandering/schooling	[-] 0.01
2.	Seek/pursuit	[-] 0.03
3.	Consuming food	[+] 100
4.	Exposed to pollution	[-] 0.05
5.	Dead fish	0

3.17. Dead Fish Condition

Dead fish objects occur when energy is at level 0. This can be caused by fish wandering without getting food, or it can be caused by exposure to pollution in the aquarium.

In the dead condition, the fish will change position by facing the surface of the aquarium, turn whitish, not reacting, not moving, then float on the surface and disappear. The death of fish in the aquarium will reduce the number of fish populations that exist.

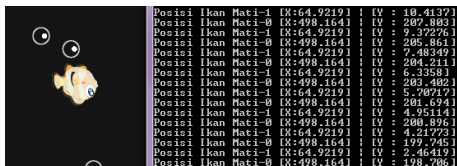


Figure 15 Change of Y coordinate position as the simulation of floating

4. TESTING

The test is conducted with several experimental processes as measurement parameters to determine the successful autonomous fish behavior system built according to the plan.

4.1 Functionality Testing of Autonomous Fish Behaviour

Functionality testing is done by testing several outputs of fish behavior towards their environment and user interactions.

Table 3

Autonomous fish Functionality test

No	Activity Description	Result
1.	Wander	Succeed
2.	Seek	Succeed
3.	Flee	Succeed
4.	Schooling	Succeed
5.	Fish sight radius factor in catching food	Succeed
6.	The ability of fish to find nearby food	Succeed
7.	Energy changes based on fish activity	Succeed
8.	Simulated dead fish floating	Succeed
9.	Simulation of gravity on worms as fish food	Succeed
10.	Implementation of collision intersection in a simulation of fish-eating worms	Succeed

4.2 Testing of the Autonomous Fish Attributes Used

Attribute testing is carried out on attributes such as energy and fish speed. Testing is done through direct observation of the application.

Table 4

Autonomous fish attribute test

No	Specification	Activity	Result
1.	Energy	Wander	Succeed enough
		Seek	Succeed
		Flee	Succeed

No	Specification	Activity	Result
		Schooling	Succeed
		Consuming food	Succeed
		Exposed to pollution	Succeed
		Dead fish	Succeed
2.	Velocity	Wander	Succeed
		Seek	Succeed enough
		Flee	Succeed enough
		Schooling	Succeed
		Consuming food	Succeed
		Exposed to pollution	Succeed
		Dead fish	Succeed

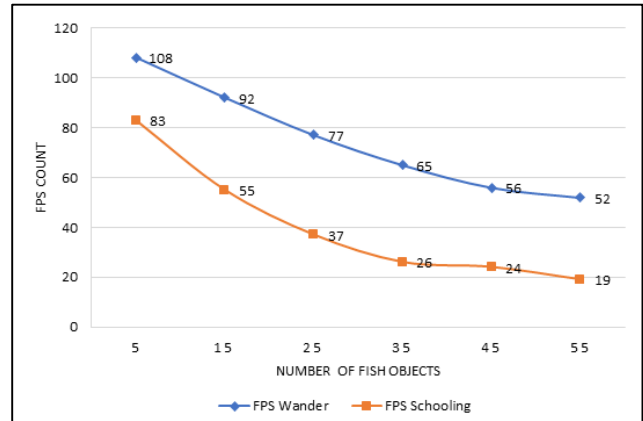


Figure 17 FPS size comparison chart for wandering and schooling behavior

4.5 FPS Testing of Catching Behaviour (Seek) Based on Number of Food Objects

The test was carried out in an application that did not include gestures, with 15 fish objects under two conditions, namely generating food simultaneously with an insignificant time lag and the second with the condition of not giving food together with a certain time lag.

4.3 Wandering Behaviour FPS Test Based on Number of Fish Objects

The next test is a test on FPS (frames per second) based on the number of fish objects used in the application. FPS is the number of frames shown per second in a moving image.

The test is carried out within ± 10 seconds with an average FPS taking.

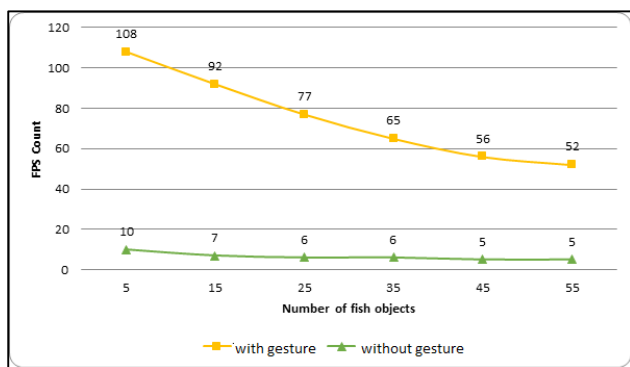


Figure 16 Comparison of FPS wander behavior based on the number of fish objects.

4.4 FPS Comparison of Wandering and Schooling Behaviour Based on Number of Fish Object

In this test, a comparative analysis was conducted on the two behaviors, wandering and schooling.

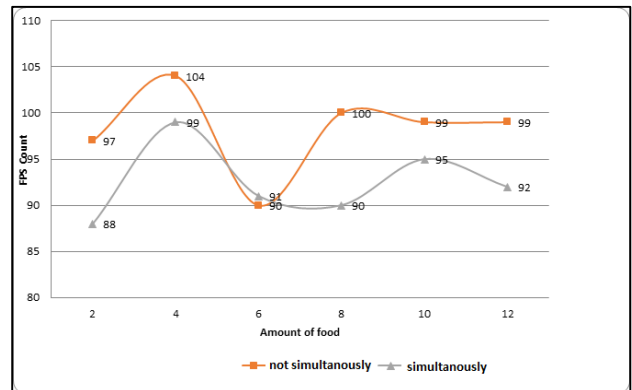


Figure 18 Comparison of the number of FPS seek behavior based on the amount of food.

4.6 Comparative Testing of Catching Behaviour with Seek and Pursuit Technique

The tests will use two techniques of reaching foods, seek steering and pursuit steering. The seek technique is commonly used in static targets, and the pursuit technique is used in dynamic targets.

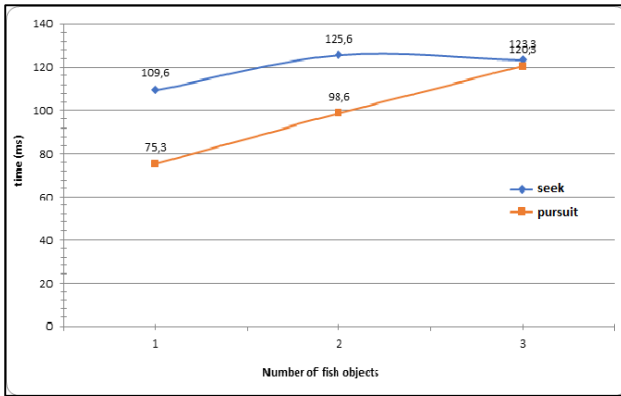


Figure 19 Seek and pursuit methods of food achievement

4.7 Result Analysis

In the testing process, several behaviors of autonomous fish that have been built can run well according to the design, both in functionality and performance.

From the FPS results, wandering behavior is proportional to the number of fish objects added. The more fish objects there are, the less the FPS is. The percentage drop in FPS is around 7% to 16%. A significant decrease occurred in applications that included gestures.

In both wandering and schooling behavior with the same number of fish, there is a significant difference in the size of the FPS. Wander behavior has a larger FPS than schooling behavior. It can be assumed that there is a high computational rate in the schooling behavior process, so that it requires higher memory than the wander process. The decline occurred with an average of almost 50% of wander behavior.

Changes can be observed in the feeding process when food is generated simultaneously and generating food is not simultaneously with a time interval. Both conditions have a significant change in FPS size. This is possible because the simultaneous generating process on food objects requires more memory, especially in the process of calculating the nearest food catch by fish that is carried out simultaneously.

In observing the behavior of catching between the seek and pursuit techniques, it can be concluded that the pursuit technique has a faster achievement time than the seek technique. Observations also show that the seek technique has a weakness in achieving targets that are always changing so that sometimes it requires several movements to reach the target. Another weakness appears in the occurrence of gathering fish objects in certain areas. The use of the pursuit technique is more appropriate to implement, considering the dynamic nature of food objects, and this method has been able to predict the target's movement.

5. CONCLUSIONS

Conclusions that can be drawn from this research include the following:

1. The use of the finite state machine method is very effective in modeling the behavior of autonomous characters. The characters in the FSM are modeled in the form of states that can be easily controlled on the existing state flows.
2. The number of fish objects used in the Interactive Aquarium application will greatly affect the FPS size. The more fish objects are used, the FPS size will decrease. The percentage of decline that occurred ranged from 7% to 16%.
3. The amount of feeding to the fish will not affect the size of the FPS. Changes in the size of the FPS only occur at the same time feeding where they require more memory.
4. Test with gesture and some fish objects decrease FPS significantly with a percentage decrease of around 90%.
5. In schooling behavior, fish have a smaller FPS than wandering behavior. The average percentage drop is almost 50%. This is because the schooling process requires higher computation.
6. The use of the pursuit technique for catching food by fish is more effective than seek technique.

REFERENCES

- [1] Aristi, Dian P F, Ari Purbayanto, Mulyono Baskoro, Takafumi Arimoto, "Ketajaman Penglihatan Ikan Juwi", Buletin PSP, vol. XI, no. 1 April. Departemen Pemanfaatan Sumberdaya Perikanan, Fakultas Perikanan dan Ilmu Kelautan. Institut Pertanian Bogor, 2010.
- [2] Buckland. Mat, Programming Game AI by Example. Wordware Publishing, 2005.
- [3] Fujaya. Y, Fisiologi Ikan (Dasar Pengembangan Teknologi Ikan). Jakarta : Penerbit Rineka Cipta, 2004.
- [4] Indrayana. Ryan K, "Rancang Bangun Zoopedia : Perilaku Non Playable Character (NPC) Untuk Simulasi Aktifitas Harian Rusa". Tesis, Magister Teknik Elektro ITB, Bandung, 2011.
- [5] Kogler. Marco, "Simulation and Visualization of Agents in 3D Environments", Diploma Thesis, Universitat Koblenz Landau, Jerman 2003.
- [6] Mismail. Budiono, Akuarium Terumbu Karang. Jakarta: UB Press, 2010.
- [7] Mulyadi. Eris, Pengenalan Tingkah Laku Ikan dalam Hubungannya Dengan Penangkapan. Semarang : Riomma, 2006.

- [8] Norvig. Peter, *Artificial Intelligence – A Modern Approach*, Prentice-Hall, Englewood Cliffs, New Jersey, 1995.
- [9] Pamungkas. Wijanarko S, “Implementasi Automated Path dan Route Finding Untuk Unit Behaviour Pada Real-Time Strategy Game Dengan Menggunakan Fuzzy Logic”, Laporan Tugas Akhir, Jurusan Teknik Informatika, Fakultas Teknologi Informasi ITS, Surabaya, 2010.
- [10] Reynolds. Craig W, “Steering Behaviours For Autonomous Characters,” Sony Computer Entertainment America, California.
- [11] Syah. Muhibbin, “Pembelajaran Aktif, Inovatif, Kreatif, Efektif dan Menyenangkan (PAIKEM)”, Pendidikan dan Latihan Profesi Guru Rayon Fakultas Tarbiyah dan Keguruan UIN Sunan Gunung Djati Bandung, 2009.
- [12] Wulandari. Ossy D E, “Perancangan dan Implementasi Teknik Steering Behaviour Pada Simulasi Makroskopik Keramaian Ritual Tawaf”. Tesis, Magister Teknik Elektro ITB, Bandung, 2009.