Flocking Boids with Geometric Vision, Perception and Recognition

James Holland
Department of Computer Science
University of Colorado at Colorado Springs
Colorado Springs, CO, 80933-7150
jholland@uccs.edu

Sudhanshu Kumar Semwal
Department of Computer Science
University of Colorado at Colorado Springs
Colorado Springs, CO, 80933-7150
semwal@eas.uccs.edu

ABSTRACT

In the natural world, we see endless examples of the behavior known as flocking. From the perspective of graphics simulation, the mechanics of flocking has been reduced down to a few basic behavioral rules. Reynolds coined the term Boid to refer to any simulated flocking, and simulated flocks by collision avoidance, velocity matching, and flock centering. Though these rules have been given other names by various researchers, implementing them into a simulation generally yields good flocking behavior. Most implementations of flocking use a forward looking visual model in which the boids sees everything around it. Our work creates a more realistic model of avian vision by including the possibility of a variety of geometric vision ranges and simple visual recognition based on what boids can see. In addition, a perception algorithm has been implemented which can determine similarity between any two boids. This makes it possible to simulate different boids simultaneously. Results of our simulations are summarized.

Keywords
Flocking simulation, geometry, Simple perception and recognition algorithms.

1. INTRODUCTION

Flocking is a method for generating animations that realistically mimic the movements of animals in nature. The most common ones we think of are birds and fish. However, there are many other examples, such as ant colonies, animal herds, and even human pedestrians. This phenomenon has been observed and studied by various scientists for many years. From a biological or sociological view the main question has been "Why do flocks form." Our main question has been "How do flocks form and how can we simulate it." Whether it is a school of mackerel, flock of geese, a herd of wildebeest, or protesting crowd moving through the streets, we must understand the factors that create and maintain these formations and then create algorithms that apply the factors. From a biological view, flocking, schooling and herding behaviors arise from a variety of motivations. By forming large groups, the average number of encounters with predators for each member of the group is reduced. It also allows predatory groups to overpower larger prey and to control large groups of smaller prey [Shaw70]. Biologists have found that there is a balance between both the need for members to be a part of the group and the need to maintain an individual space, which cause groups to form [Vehe87]. The concept of attraction versus maintaining space has been the focus of much of the research in simulating animations of flocking behavior to date. In 1987, Reynolds described a flock as the result of the interaction between the behaviors of individual birds [Reyn87]. To simulate a flock we need to simulate the behavior of each individual bird. Once the basic mechanics of motion for the boid is defined, the behavioral model boils down to three rules:

- Collision Avoidance or Separation: avoiding collisions with flock mates.

- Velocity Matching or Alignment: matching velocity and direction with nearby flock mates.
- Flock Centering or Cohesion: attempting to stay near the center of the flock.

Since Reynolds's inaugural research into flocking behavior, others have extended his work in various directions adding actions such as pursuit and evasion, wandering, exploring, homing, and shepherding [Reyn99]. Anderson et al. extended the path following with the idea of constrained group animation [Ande03]. Brogan and Hodges implemented a visual model in which the number of visible neighbors affect how separation, cohesion, alignment and other factors are applied to decision making [Brog97]. Jadbabaie et. al. experimented with the nearest neighbor flocking algorithm in which data from only one flock mate is used to modify a boids path [Jadb03]. Other researchers have explored the way boids think to create more realistic animations. Bajec et al. introduce fuzzy logic into the decision making process of the boids [Baje03]. Pedestrians and crowd simulation is in [Shao07, Sull02]. Predefined paths are used in [Ande03]. By applying global environmental knowledge, Bayazit et al. implemented three new group behaviors: homing, exploring and shepherding [Baya02]. Hartman and Benes introduce the concept of leadership into the flock in order to create a more realistic looking behavior [Hart06]. Musse et al. applied flocking rules to implement a Crown Model for Virtual Environment [Muss98].

2. MOTIVATION: BOIDS WITH EYES

The inspiration for our work came while searching for information about bird flocking. One web page that came up in the search showed the following images [Noll07] in Figure 1.

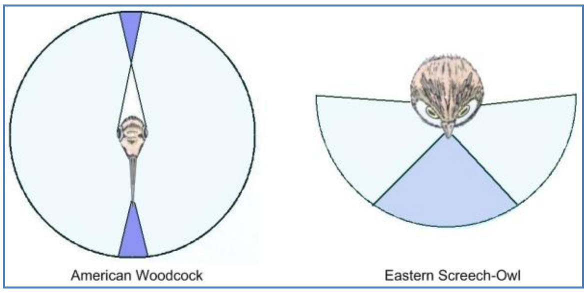

Figure 1. Avian field of vision.

Seeing these images and knowing that owls do not flock, and that woodcocks do, made us ask this question. Is flocking behavior related to vision? If it is, then how is it related?

As the concept of visual perception in flocking had not been explored especially in the context of Figure 1, we decided to explore how the boids visually perceive their environment and how the information gathered could be used to influence their decision making process.

We started by considering vision in general. There are two sides of the vision process. First there are the physical aspects of vision. These attribute are based on some physical attribute of the eye, either its geometry or its light receptors. They define the geometric field in which the eye can see, the range of light and changes in light the eye can detect, and how the eye can adapt and change to changes in the environment to affect what is seen. These aspects include, but are not limited to

(a) Light Reception Based -- Acuity: Clearness or acuteness of vision. Spectral Response: Range of light wavelengths the eye can detect. Dynamic Range: The eyes ability to adapt to changes in light. Accommodation: The eye's ability to adjust its optical power in order to maintain a clear image or focus objects at various distances. Resolution: Minimum size detail the eye can detect or distinguish.

(b) Geometric -- Visual Field: Angular span of vision. Perceptual Span: Angular span of vision in which the eye can accurately focus. Eye Placement: How and where the eye is positioned in the head. Peripheral Vision: Vision that occurs outside the very center of gaze. Binocular Vision: Vision in which both eyes are used together to produce depth perception. Eye Movement: Ability to move within the eye socket.

The other important aspect of the vision process is the visual perception. In psychology, visual perception refers to the process of acquiring, selecting, and interpreting visual sensory information.

After researching vision further, it was decided that the implementation would be divided into three parts. First, the physical aspects of vision will be defined by a geometric model of bird vision. Next perception algorithms will gather and interpret the data provided by the model. Finally the interpreted data will be applied to the flocking decision processes. We also expanded the topic to include pattern recognition during the simulation of the perception algorithm. In order to accomplish this, each boid would need to include a model of its physical appearance that the other boids could perceive.

3. THE VISUAL MODEL

The visual model needs to model the geometric aspects of vision discussed earlier as closely as possible. It needs to adapt easily to match the visual characteristics of a variety of birds. And finally, it needs be designed in a way that can be implemented into the perception algorithms, the flocking algorithms, and the graphical view. To satisfy these requirements, the model will have four parameters derived from the attributes of vision and the Owl and Woodcock images in the previous section, as follows: (a) Eye position angle (b) Direction of vision (c) Field of vision (d) Range of vision. The Eye Position Angle describes the placement of the eyes on the boid's head. As seen in Figure 2, in birds the placement of the eyes varies between species. They can be in the front of the head as in the Owl or on the side as in the Woodcock. The vertex of the angle is at the center of the head and is relative to the forward direction. At the minimum angle, zero degrees, the eyes appear on the front of the head facing directly forward. At the maximum angle, 90 degrees, the eyes appear on the side of the head, at a right angle to the forward direction.

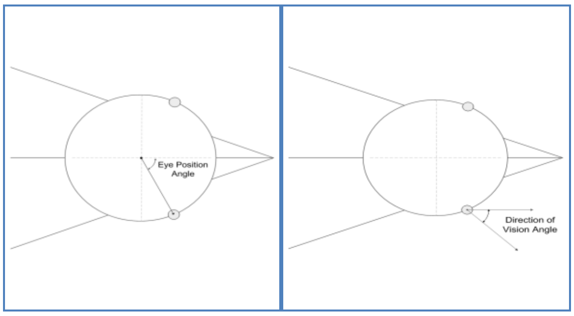

Figure 2. Eye position angle (Left) and Direction of Vision (Right).

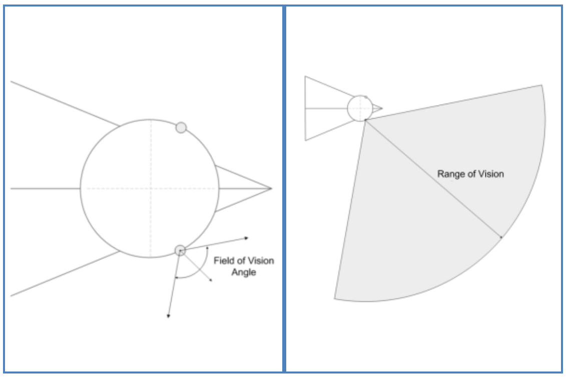

Figure 3a. Field of vision (Left) and range of vision (Right).

The field of vision defines the limits of vision both horizontally and vertically. Our model will not distinguish between the two, but uses a single value creating a circular cone of vision. As with the direction of vision, the field of vision vertex is at the center of the eye, see Figure 3a(left), and its angle is relative to and symmetrical about the direction of vision. The Range of Vision defines the maximum distance which the bird can recognize objects. The range is measured from eye radiating out in the direction of vision. As seen in Figure 3a(right), the range of vision specifies the height of the conical visual field.

When combined in three dimensions, the visual parameters create a circular conical visual field that emanates from each eye. This visual field mimics the visual abilities of birds and can be easily modified to match a variety of bird visual models. Figure 3 (b) shows Boids with eyes used in our implementation. They appear as dots in simulations as shown later.

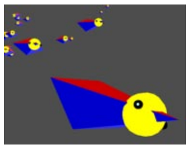

Figure 3b. Boid with eyes.

4. PATTERN RECOGNITION MODEL

A model for pattern recognition for birds is not completely known; however we do know birds have very acute vision. Their eyes have five times as many receptors as human eyes. Some birds of prey can track a rabbit from a mile away. The pattern model we chose to represent the boids is a simple bit pattern. Each boid is represented by a set of bits with a length from 1 to 32 bits. Each bit or set of bits would represent some attribute about the boid, such as color, wing shape, etc. The comparison algorithm performs a bit by bit or bit set by bit set comparison.

4.1 The Perception Algorithm

The purpose of the Perception Algorithm is to gather information from boids environment base upon its visual model. The algorithm first determines which neighbor boids it can see. It then perceives the following information about the visible neighbors: location, direction of travel, speed, and appearance.

The perception algorithm is broken into three phases: view, perspective, and pattern recognition.

View Phase

The View Phase creates the initial visible neighbors list and uses the field of vision and range of vision to eliminate objects that are outside the visual gield of the boid. The first part of the algorithm calculates the distance to each neighbor and compares it to the Range of Vision. If the neighbor is beyond Range of Vision it is removed from the list. The second step calculates the angle to each neighbor relative to the direction of vision for both eyes. These angles are then compared to the field of vision angle to determine if the neighbor is within the Boid's visual field. If the angle is less than half the Field of Vision angle then the boid is visible. Following this phase, each neighbor boid will be marked as either not visible, visible from either the left eye, right eye or both eyes. Figure 4 below shows two object near a boid. Both object are within the range of vision, however only Object A is in view. The angle to Object B is greater than half of the Field of Vision.

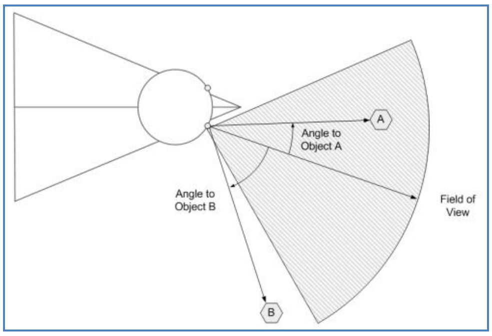

Figure 4. View phase.

Perspective Phase

In the perspective phase, the algorithm searches through the list of neighbors looking for neighbors which are obscured by closer neighbors. It starts by sorting the list by distance from the boid. Starting with the closest neighbor the algorithm checks each further neighbor to determine if the closer neighbor obscures the further one. Obscured neighbors are marked as they are found. Partially obscured neighbors are considered visible.

Obscured Object Culling

First the algorithm projects the neighbors' bounding circles onto the boid's unit sphere. This is done by dividing the radius of each object by its distance from the boid.The algorithm then calculates the distance between the projected centers of the neighbors, see Figure 5a. We first project the centers onto boid's unit sphere. It then gets the offset vector between the projected centers. The length of this vector is the distance between the projected centers. In the comparison phase, the projected radii are compared to determine if the further object is obscured. There are four possible outcomes of the comparison:

1. If the further object's radius is greater than the closer object's radius, the further object is at least partially visible.
2. If the distance between the projected circles centers is less than or equal to the difference in their radii, see Figure 5b, then the further object is completely obscured.
3. If the distance between the projected circles centers is greater than or equal to the difference in their radii, see Figure 5c, then the further object is partially visible.
4. If the distance between the centers is greater than or equal to the sum of the radii, see Figure 5d, then the further object is completely visible.

For our implementation, a neighbor is considered obscured only if it is completely obscured.

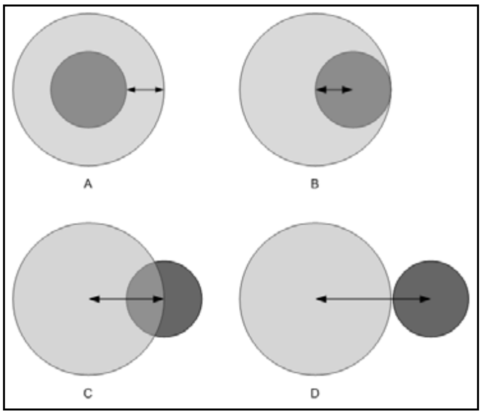

Figure 5. Obscured object culling.

Pattern Recognition Phase

In the Pattern Recognition phase of the Perception Algorithm, the pattern of each visible neighbor boid is compared to the boids own pattern. It is assumed that if the neighbor boid is visible then the pattern can be discerned. The pattern match algorithm counts the number of bits that are similar between the boid and each neighbor. An exclusive-OR algorithm can determine this count. This count (factor) is stored and is used in the flocking/steering algorithms.

5. OpenSteer LIBRARY

The flocking and steering phase of the program is based upon the algorithm developed by Craig Reynolds in his OpenSteer Library. The OpenSteer C++ library contains three algorithms for flocking simulations: steer for cohesion, steer for separation, and steer for alignment. We created two modifications version of each of these algorithms that employ the data derived by the perception algorithm to influence steering decisions. The first version employs the visibility and perspective variables to limit the neighbors used in the steering calculation to those that are visible. The second version applies a pattern recognition based weighting to the decision process.

 ISBN 978-80-86943-93-0

Steer for cohesion

The Steer for Cohesion algorithm keeps the flock together. It does this by applying steering forces to the boid that drive it towards the flock's center. The general algorithm calculates the center of all the neighboring boids and applies a steering vector toward this point. In the modified steer for cohesion algorithm, only boids that are visible are used to calculate the flock center. The algorithm calculates the flock center by averaging the positions of all visible neighbors. The steering vector is the normalized difference between the boid's position and the flock center. When our pattern recognition is applied, the steering vector is weighted based on the pattern match value (factor) calculated in the Perception Algorithm. Because each neighbor may have a different pattern and therefore require different weighting, the pattern recognition version cannot look at the visible neighbors as a whole, but must calculate steering vectors for each individually. The final steering vector is the normalized sum of the steering divided by the sum of the pattern weighting.

Steer for separation

Then steer for separation algorithm keep boids from running into each other. In the basic steer for separation algorithm, the boids are steered away from their nearest visible neighbors. The algorithm calculates a steering vector for each neighbor that is opposite of the offset vector between the two boids. The steering vector is divided by the squared distance between the boid to get the 1/distance falloff. This causes the magnitude of the vector to drop off as the distance between the boids increases. When our pattern recognition is applied, each neighbor's position relationship is weighted based on the pattern match value.

Steer for alignment

In the steer for alignment algorithm, the boids are turned toward the visible flock's predominant direction. The algorithm calculates the flock direction by averaging the forward vector of all visible neighbors. The steering vector is the normalized difference between the boid's forward vector and the flock's. When our pattern recognition is applied, each neighbor's alignment relationship is weighted based on the pattern match value. The Boid's visual model defines its physical attributes of vision and its perception algorithms perceive and interpret its environment. The information gathered then influences the Boid's decisions.

6. IMPLEMENTATION DETAILS

The Simulation Mode generates the real-time flocking simulations. Sequence of functions are called during each update cycle. The processing is broken into two phases, update and draw. During the first phase, calls to the pluginUpdate and boidUpdate functions generate new steering vectors and in turn new position and orientation vectors for each boid in the flock. The second phase calls to pluginRedraw and boidDraw functions to display the boids using the newly generated vectors. To facilitate the analysis process, we used an automated script to generate simulation files for all combinations of the following parameters and values:

- Normal Mode and Pattern Recognition Mode
- Flock Size: 20, 50, 100
- Direction of Vision: 0, 20, 40, 60, 80, 90
- Field of Vision: 30, 60, 90, 120, 150, 180
- Range of Vision: 10, 20, 40

The script generated 648 simulation files. During the generation process, the application's analysis functions, extracted simulation and flock data sets for statistical analysis. Details of statstical analysis is presented elsewhere [Holl07]. Here we present the main results.

Flock Simulation Analysis

In this section, we will subjectively analyze four simulations out of a total of seven simulations we perfomed in [Holl07]. Each simulation is with different flock and visual settings. We will characterize the simulation with the following criteria: flock alignment, flock cohesion, and stray boids. In each simulation, we took screen shots once a second starting 3 seconds after simulation initialization. The three seconds allows the boids to form the initial flock. The number of screens taken depends on how long the simulation continued to generate interesting results. We have also perfomed studies where pattern recognition is activated (Sample 4) . In sample 4 simulation, there are four types of boids visible. The table below show the data for each pattern used in the simulations. As described in the design and implementation [Holl07], the patterns affect how the boids react to each other based on the pattern match value.

ID	Pattern	%of Boids	Body Color	Head Color
Boid 1	00000000	35	Yellow	Orange
Boid 2	00111010	35	Green	Red
Boid 3	11000101	15	Blue	Yellow
Boid 4	11111111	15	Red	Magenta

Table 1. Boid patterns.

Simulation example -- Sample 1

Simulation 1 (Figure 7) contains 100 boids with the following visual settings:

- Direction of vision: 20 degrees.
- Field of vision: 60 degrees.
- Range of vision: 20 units.

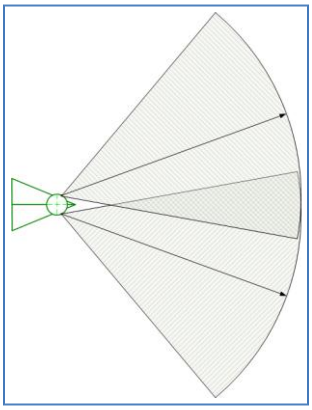

Figure 7. Simulation 1 Boid View.

The images in Figure 8 show the boids quickly moving away in random directions. By the seventh second, it is apparent that the initial flock will scatter and that there is no flock cohesion or alignment.

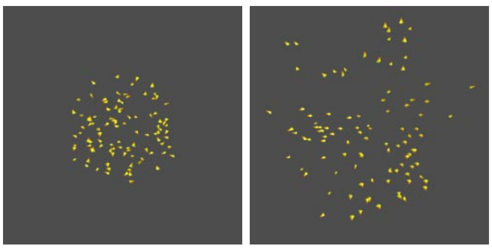

Figure 8. Boids move in random direction in Left: 3 seconds, and Right: 11 seconds.

Simulation example -- Sample 2

Next we started 100 boids with the following visual settings:

- Direction of vision: 90 degrees.
- Field of vision: 60 degrees.
- Range of vision: 20 units.

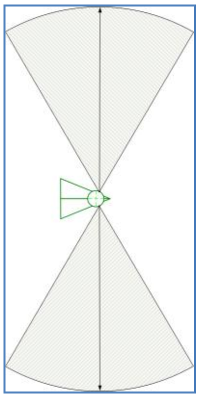

Figure 8. Simulation 2 Boid View.

The images in Figure 9 shows that within the first few seconds, a cohesive flock has formed. The flock maintains cohesion and alignment throughout the six seconds of the simulation displayed.

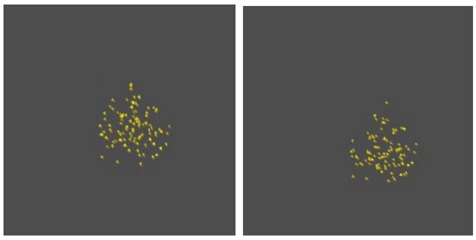

Figure 9. Boids move in random direction in Left: 3 seconds, and Right: 7 seconds.

Simulation example -- Sample 3

The final simulation again contains 100 boids with the following visual settings:

- Direction of vision: 90 degrees.
- Field of vision: 200 degrees.
- Range of vision: 20 units.

These parameters match those of the Woodcock presented earlier. The image below, Figure 10, show the field of vision the parameters define.

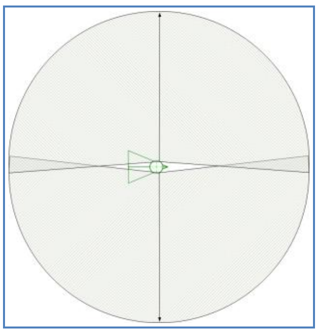

Figure 10. Simulation 3 Woodcock View.

ISBN 978-80-86943-93-0

The flock illustrated by the images in Figure 11 maintains a consistent shape and size throughout the ten-second simulation. Alignment and cohesion are constant through every image.

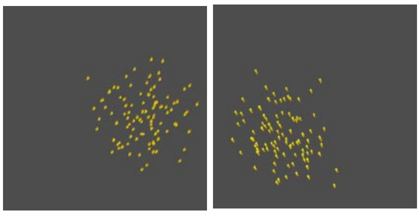

Figure 11. Boids move in random direction in Left: 3 seconds, and Right: 11 seconds.

Simulation Example -- Sample 4

Simulation 4 contains 100 boids with the following visual settings:

- Direction of vision: 20 degrees.
- Field of vision: 180 degrees.
- Range of vision: 20 units.

As seen in Figure 12 below, the boids in this simulation have a field of view that extend 110 degrees left and right.

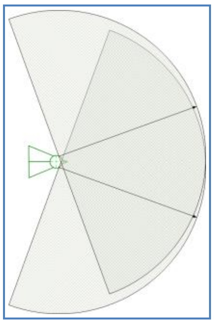

Figure 12. Simulation 4 Boid view.

In this simulation, the affect of the Pattern Matching is obvious and significant. Within one and a half seconds (Figure 13 left), the colors are coalescing and at three seconds (Figure 13 right), the colored flocks are steering away from each other. Pattern cohesion, alignment, and separation are all visible.

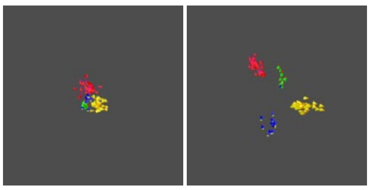

Figure 13. Different boids start to coalesce (left,1.5 seconds) and then steering away (right, 3 seconds).

7. RESULTS

Early on it in our simulation it became apparent to us that side vision is significant for flocking behavior. From the Direction of Vision Summary, we see that in the first few seconds of a simulation, boids with no side vision do not visually perceive their neighbors and the flock quickly disperses. We also see a general instability in the flock data for boids that have more forward-looking vision.

We performed several other simulations to confirm this observation. Simulations having a field of vision of 60 degrees does not seem to be a wide enough field of view to afford flock cohesion, however as we compared the three simulation with direction of vision moving from forward to a more lateral view, cohesion and alignment improve.

From the field of vision statistics, we observed that the greater the field of vision the more stable the flocking. This result was expected. This is also confirmed from field of vision statistics summary, which can be found in [Holl07]. The graphs all indicate that the flocks characteristic improve as the field of vision widens. They also indicate that when the field of vision drops below 90 degrees flock cohesion breaks down and the flock will disperse.

We also observed that boids with narrower fields of vision tend to fly fast and form elongated flock, is due to a phenomenon which we have termed as the *leapfrog affect* [Holl07]. The affect is most noticeable when the direction of vision is near zero. In simulations of boids with this visual model, the boids form into flocks that are near single file lines. As the flocks move, boids will leapfrog past the boids in front of them moving to the front of the flock. This leapfrogging continues as the flock moves through space.

The cohesion steering algorithm causes the leapfrog affect. For boids at the back or middle of the elongated flock, the center of the flock, which they are drawn towards by the cohesion algorithm, is a point half way between them and the lead boid. The cohesion steering causes them to accelerate toward that point. The acceleration leapfrogs them past the boids directly ahead of them. Once they reach a point near or at the front of the flock the cohesion affect lessens and they slow to match the flocks speed. Simulation motivated from the woodcock image, generated the expected results. Boids that can see everything around them have no problem forming and maintaining a cohesive flock. Similarly, simulation motivated from the owl image, generated the expected results as well. The owl flock did not

 ISBN 978-80-86943-93-0

have the stability and compactness observed in the woodcock simulation.

8. CONCLUSIONS AND FURTHER RESEARCH

We can now revisit the following questions we posed earlier. Is flocking behavior related to vision? If it is, then how is it related? Based on the simulation generated, the answer to the first question is yes. The analysis shows correlations between the visual models and the flocking behaviors observed. Boids with narrow forwards looking vision could not form flocks. As the field of vision is expanded, the flocks formed are elongated and exhibit odd behaviors. The boids with visual models with direction of vision that is more sideward than forward and fields of vision that extend to give the boid backward view formed stable flocks that exhibit behavior resembling those of real birds. There are several aspects of vision that the current model and algorithms do not implement—(i) Binocular vision and depth perception;(ii) Accom modation and motion-perception; (iii) Peripheral vision and vision impairments (iv) environmental aspects; (v) Visual medium variations; (vi) Predator/prey relationships and behaviors; (vii) Obstacles.

There are several areas related to the Pattern Recognition that could be explored – (i) Obscure portions of the pattern based the boids perspective; (ii) Distort the pattern based on distance or environmental conditions; (iii) The modeling concept could be extended to other boid attributes; (iv) Flight Characteristics of Birds(v) Swimming Characteristics of Fish; (vi) Other Senses; (vi) Cognitive Abilities.

From the simulations it is also apparent that the visual model is not the only factor that determines whether a species of birds forms flocks. It may determine whether a bird can form a flock, but not all birds that have visual models that according to this work should be able to form flock, do.

9. REFERENCES

[Ande03] Anderson, M., McDaniel, E. and Chenney, S. Constrained animation of flocks. Eurographics/SIGGRAPH Symposium on Computer Animation, pp. 286-297, 2003.

[Baja03] Bajec, I., Lebar, M. Mraz, N.Z. Boids With A fuzzy way of thinking. Proceedings of ASC, pages 58-62, 2003

[Baya02] Bayazit, O., Burchan, JML., and Amato. NM. Roadmap based flocking for complex environments. Proceedings of the 10th Pacific Conference on Computer Graphics and Applications, 2002

[Brog91] Brogan, DC. And Hodgins, JK. Group Behaviors for Systems with Significant Dynamics. Autonomous Robots, 4, 137-153, 1997.

[Hart06] Hartman, C., Benes, B. Autonomous boids. Computer Animation And Virtual Worlds 17, pp. 199–206, 2006

[Holl07] Holland, J. Flocking boids with visual modeling and pattern recognition. MS Thesis University of Colorado, Colorado Springs, Advisor: SK Semwal, pp. 1-111, 2007.

[Jadb03] Jadbabaie A., Lin, J., and Morse, A.S. Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Transactions On Automatic Control, Vol. 48, No. 6, June 2003

[Muss98] Musse, S.R., Babski, C., Capin, T., Thalmann, D. Crowd modeling in collaborative virtual environments. In VRST'98: Proceedings of the ACM Symposium on Virtual Reality Software and Technology. ACM Press: New York, pp. 115–123, 1998.

[Shao07] Shao, W. and Terzopoulos, D. Autonomous Pedestrians. Graphical Models, 69(5-6), September November, pp. 246-276, 2007.

[Sull02] Sullivan, C.O., Cassell, J., Vilhjalmsson, H., Dinglianna,J., Dobbyn, S., McNamee, B., Peters, C., and Giang, T. Levels of details for Crowds and Groups, Computer Graphics Forum, 21(4), 2002.

[Reyn87] Reynolds, C.W. Flocks, Herds, And Schools: A Distributed Behavior Model. In Computer Graphics: SIGGRAPH '87 Conference Proceedings, volume 21(4), pp. 23-34. ACM SIGGRAPH, 1987.

[Reyn99] Reynolds, C.W. Steering Behaviors For Autonomous Characters. In 1999 Game Developers Conference, pp. 763-782, 1999

[Shaw70] Shaw, E., Schooling in fishes: critique and review. Development and Evolution of Behavior. L. Aronson, E. Tobach, D. Leherman, and J. Rosenblatt (eds), W. H. Freeman: San Francisco, CA, pp. 452—480, 1970.

[Vehe87] Veherencamp, S., Handbook of Behavioral Neurobiology, Volume 3: Social Behavior and Communication, P. Marler and J. G. Vandenbergh (eds.), Plenum Press: New York, NY, pp. 354-382, 1987.

[Noll07] Noll, P. Sense of vision for birds. Image. www.paulnoll.com/Oregon/Birds/Avianvision.html, 2007.