Visual Deformation by Swarm – a Technique for Virtual Liquidizer of Objects

Prof. T. Unemi, DEng.
Department of Information Systems Science, Soka University, Hachioji, Japan
http://www.intlab.soka.ac.jp/~unemi/
e-mail: unemi@iss.soka.ac.jp

Dr. D. Bisig, PhD.
Zurich University of the Arts, Institute for Computer Music and Sound Technology, Zurich, Switzerland
http://bitingbit.org/intro.php
e-mail: daniel.bisig@zhdk.ch

Premise

This paper presents a method to apply a type of swarm simulation to generate interesting patterns of deformation that provides an experience to visitors as if their bodies were liquidized. Utilizing a combination of two different types of swarm formation algorithms, BOIDS and ANT; it is possible to focus the deformation only on the part of visitors' bodies captured by live cameras. This mechanism organizes a distribution of 2D vectors on the display area. The final image on the display is rendered using an interpolation algorithm that generates spatially smooth image in any resolution by taking an advantage of GPU power. Parallel processing by multi-core CPU is also helpful to guarantee the smooth movement and quick response for interactive installation.

1. Introduction

Swarm simulation is a useful technique to produce complex dynamic patterns as response to any type of real-time changes of environment as shown in our previous works [1-4] and many others such as [5]. The mechanism was developed as models of collective behavior of animals mainly in the researches of Artificial Life to deepen our understandings on biological complex system. The targets include herbivore, birds, fish, grasshoppers, mosquitos, ants, termites, and so on. Usually, the movement of the simulated swarm is used as a type of brush strokes to draw aesthetic patterns by computer, but we propose here a method to use it to generate a type of deformation in order to provide a virtual experience to visitors as if their bodies were liquidized.

It is possible to realize any type of deformation of a 2D image if appropriate distribution of 2D vectors over the whole area of the canvas is given. Assigning a position in the original image for each agent, such distribution that is changing dynamically is easily organized by combination with an interpolation method of continuous vector field. By starting from uniform distribution of agents each of which memorizes its starting position, the movement of swarm produces a dynamic pattern of deformation gradually changing from original to chaotic.

For our new interactive installation named *Visual Liquidizer or Virtual Merge* [6], we employed BOIDS algorithm in 2D space as the basic mechanism to design the swarm activity. It forms a type of flocking behavior similar to birds and fish by local interactions among individual agents. To restrict the area of deformation into the part of target object, namely visitor's body, we organize another type of swarm controlled by ANT algorithm that densely inhabits the target area and sparsely inhabits the background area. Each ANT agent is attracted by virtual chemicals supplied at the area, where the target is detected by background subtraction or depth information from the Kinect sensor. The chemicals gradually evaporate and diffuse to form distribution of density gradients as to summon the agents properly. Each BOIDS agent is paired with an ANT agent in one-to-one relation, and only the BOIDS agents whose partners are in the target area contribute to form the swarm. The continuous distribution of 2D vectors in the rendered area is computed based on the distribution of BOIDS agents each of which provides the position of paired ANT agent as a sample point.

To provide experience that is more impressive for visitors, we added two mechanisms; to temporarily reunite some recognizable parts of original image, and to return the scattered image perfectly back to the original one at the final stage.

The following sections describe details of behavior models of swarm, an interpolation method to organize a continuous vector field, techniques of parallel processing to accelerate the response speed, additional mechanisms of reunion and homing, and then some concluding remarks.

2. Swarm

We employ a combination of two different models of swarm behavior here, ANT and BOIDS.

ANT algorithm is a model of collective behavior of ants and termites. They live in the well-constructed nest as a group of large number of individuals. Each individual has its own role in the organization. The most individuals are the workers for nest maintenance, larva care, and foraging. The model focuses on the teamwork by workers who seek and gather foods from outside of the nest to feed all of members of the group. It is still an unsolved mystery in the field of ethology, but one of the well-known hypothetical mechanisms behind the teamwork is a communication through the pheromone, a special biochemical each individual produces and senses. In a typical ants living in the nest under ground, foraging workers start roaming from the nest every morning in almost random walks. Once a worker discovers a food, such as a dead body of another insect, she carries it back to the nest if possible. In case the object is too large to bring alone, she releases pheromone on the ground. The chemical gradually defuses and spread around the place. It organizes a distribution of density gradients gradually descending from the food position to the surrounding area. Once another food seeker detects the pheromone on the ground, she stops random roaming and starts walking directed toward higher density of the chemical. As the result of a number of workers gather together around the food, they carry it to the nest by cooperation. If the food is decomposable by individual workers to carry its small part by each, they organize a long stable line of transportation between the food site and the nest. Because the chemical on the ground gradually becomes thinner by evaporation, the line breaks after they finish carrying all of foods. This mechanism is called pheromone trail that inspired a useful method to find an optimal route in a decentralized system in the industrial applications such as packet routing in the communication lines and traffic planning in the transportation system. These application-oriented researches and developments are on going under the name of Ant Colony Optimization [7]. The team organization mechanism by pheromone is a type of communication mediated by signals recorded in the environment, but not a direct mutual conversation between individuals.

We employ ANT algorithm to follow the target area in the 2D space. The attractant chemicals are not provided by individual agent, but the image processing algorithm places a fixed amount of signal in the cell of memory lattice corresponding to the 2D position. The signal spreads to neighboring cells by taking a weighted summation of the amounts, and decreases by multiplying a coefficient less than one to simulate the evaporation. When nothing is detected as a target, that is, there is no signal in the memory lattice; the each individual agent is roaming in a random walk relatively fast. Once an agent finds a gradient of signals at the corresponding position of memory, it starts moving slowly toward the direction of higher value of signals by observing the values in the neighboring cells. By adding an auxiliary repulsion force between agents, this mechanism achieves an efficient arrangement of agents where they gather and stay in the target area in higher density than the other area as shown in Figure 1.

Figure 1. A simple positioning of ANT agents. They gather in the target area with higher density.

BOIDS algorithm is a model of collective behavior of fish, birds and herbivores. They move together with a number of individuals with neither guidance from outside nor central control by a leader, but just a simple mechanism by each member. Such type of decentralized group behavior is useful to reduce a risk of a predator's attack. By splitting a group into subgroups when a single big predator is approaching, some subgroups will survive and it avoids extinction of whole of the original group. It is also helpful for long distance migration of birds and herbivores by finding an appropriate path by some members and following them by others. The development of the model of such behavior is useful not only to understand more about animal behavior in the context of ethology but also to develop software that simulates a group of such animals for movie animation. We can find the pioneering work of computer animation of mixture of fish and birds by C. W. Reynolds in late of 1980's [8].

In this model, each agent follows three types of rules in principal by observing the other neighbouring agents surrounding it. Those are cohesion to gather, repulsion to avoid collision, and alignment to move together. With some additional parameters such as physical coefficients of mass and friction, limitation of observable area from each agent, delay of reaction, limitation of acceleration and speed of both translational and rotating movements, and so on; it is possible to develop a variety of behavior styles from mosquitos to geese.

We use this algorithm to produce a complex pattern in 2D plane. An alternative method might be a simulation of fluid and powder by particles or finite (or boundary) element methods. Those techniques seem more natural for simulation of flowing liquid than BOIDS, but here the objects to be liquidized are not simple physical entities but living things, that is, visitors' bodies. BIODS is more effective to produce unpredictable complexity which provides an illusion to the visitors as if there were something alive behind the observation.

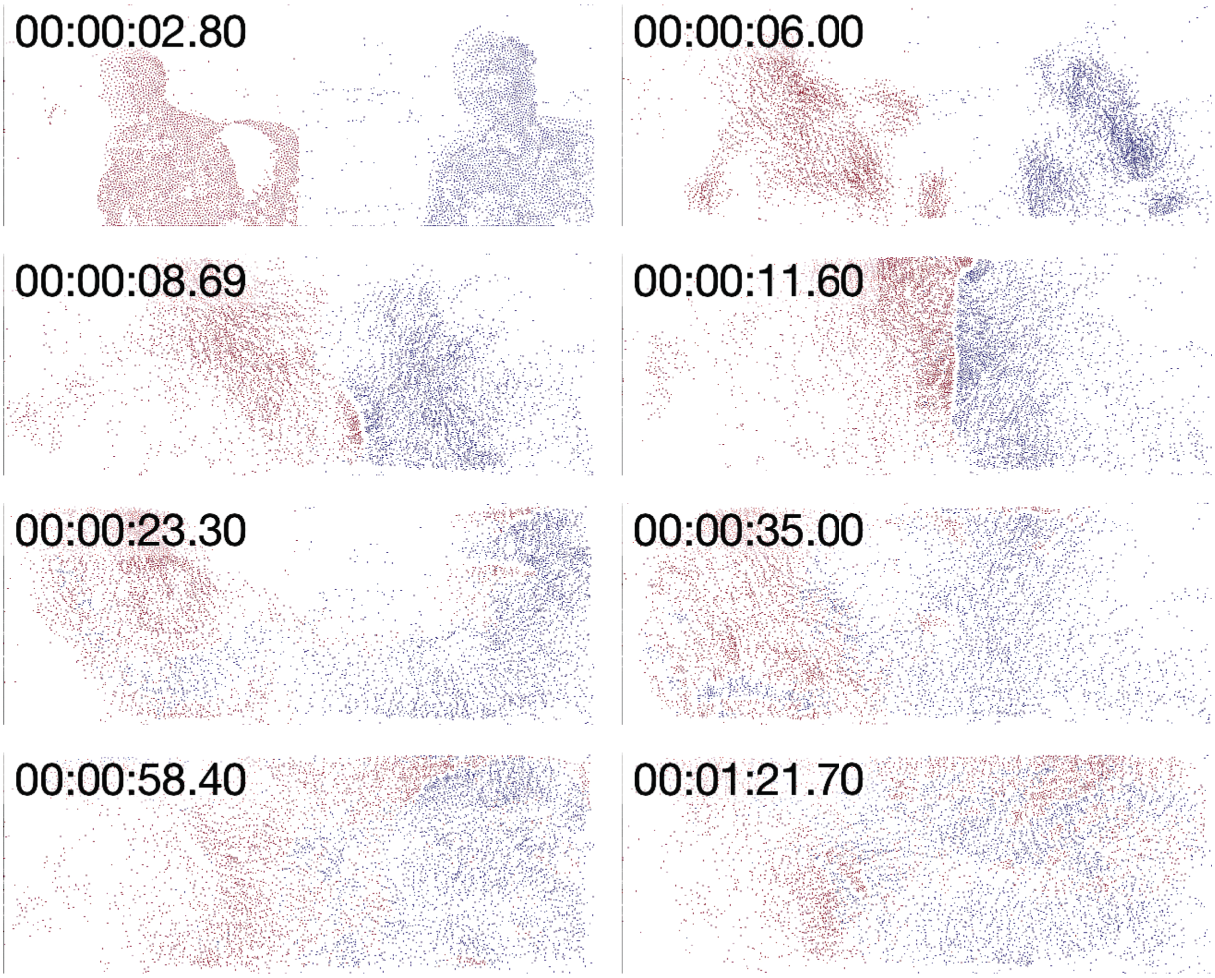

Figure 2. A sample movement of BOIDS agents in Visual Liquidizer or Virtual Merge. Blue dots are agents whose partners are at the left target and red dots are agents whose partners are at the right target. The frame sequence begins from the top left, and proceeds toward the bottom for each line scanned from left to right.

Each agent is coupled with one of the other type of agent in one to one relation, that is, the population is organized by a number of pairs of one ANT agent and one BOIDS agent for each. BOIDS agent is just carrying the 2D coordinate of coupled ANT agent's position to another position. By allowing the BOIDS agents to participate the collective behavior only if the partner settles in the target area, it is possible to produce a deformed pattern only of the target object. By assigning the same position of the partner ANT agent as the initial state, the movement of BOIDS swarm produces a dynamic deformation process that gradually changes the image from the original to chaotic. An example movement of BIODS agents is shown in Figure 2.

3. Interpolation

As described above, the arrangement of BOIDS agents provides a distribution of 2D vector expressing the position of ANT agent. We employ an interpolation method to construct a continuous distribution of 2D vectors to fill the area BOIDS agents are flocking. This distribution represents a function that maps a vector value to another vector value where both are indicating positions of 2D space. Each pixel in the final image is rendered with the color extracted from the indicated position of camera image according to this function. An alternative method is to interpolate not positions but colors. However, color interpolation produces grayish blurry image when the agents are fully mixed, though position interpolation results very complex textures.

In the interpolation method, the estimated value v_i at position p_i is calculated from a set of samples S by the following equation.

$$v_i = \begin{cases} v_i & \text{if } i \in S \\ \dfrac{\sum_{j \in S} w_{ij} v_j}{\sum_{j \in S} w_{ij}}, \quad w_{ij} = |p_i - p_j|^{-\alpha} & \text{otherwise} \end{cases} \tag{1}$$

where $| p_i - p_j |$ is the Euclidean distance between p_i and p_j, and α is a positive coefficient. The terrain of interpolated surface becomes smooth if this coefficient is larger than 2. In out application, v_i is the position of ANT agent, and p_i is the position of BOIDS agent. We use α = 1 based on our preliminary experiments for a variety of values. This setting makes pointed peeks at sampled points, but the rendered image looks more natural than the case of larger value of α even when the sample points are sparsely positioned.

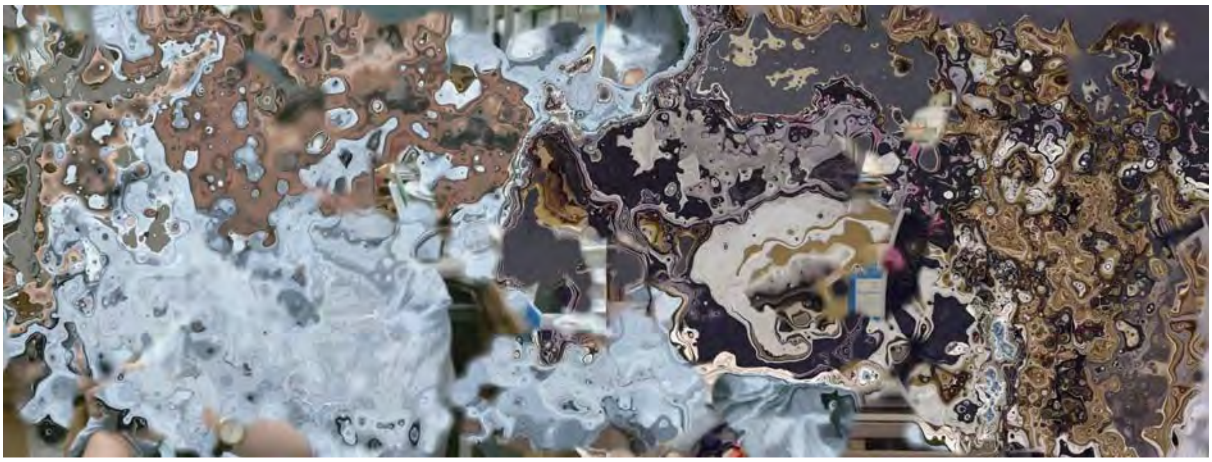

Figure 3. A sample image rendered with the interpolation algorithm from a set of distributed sample points.

To render the deformed image, it is necessary to compute the above equation for each pixel of the image unless the pixel position is far away from any sample points.

Theoretically, it requires the computational cost proportional to the number of pixels multiplied by the number of sample points. Because the target application of this algorithm is a type of real-time visual interactive installation, it is an important point whether the rendering process of one frame image finishes within 1/30 second in order to guarantee a smooth motion and quick response. The other parts of computation necessary to drive this installation are also relatively heavy as described later, but this part is the heaviest because the display of required resolution includes approximately half a mega pixels. The detail of parallel processing to reduce the computation time is described in the later section. Figure 3 shows a sample image of display generated with this algorithm.

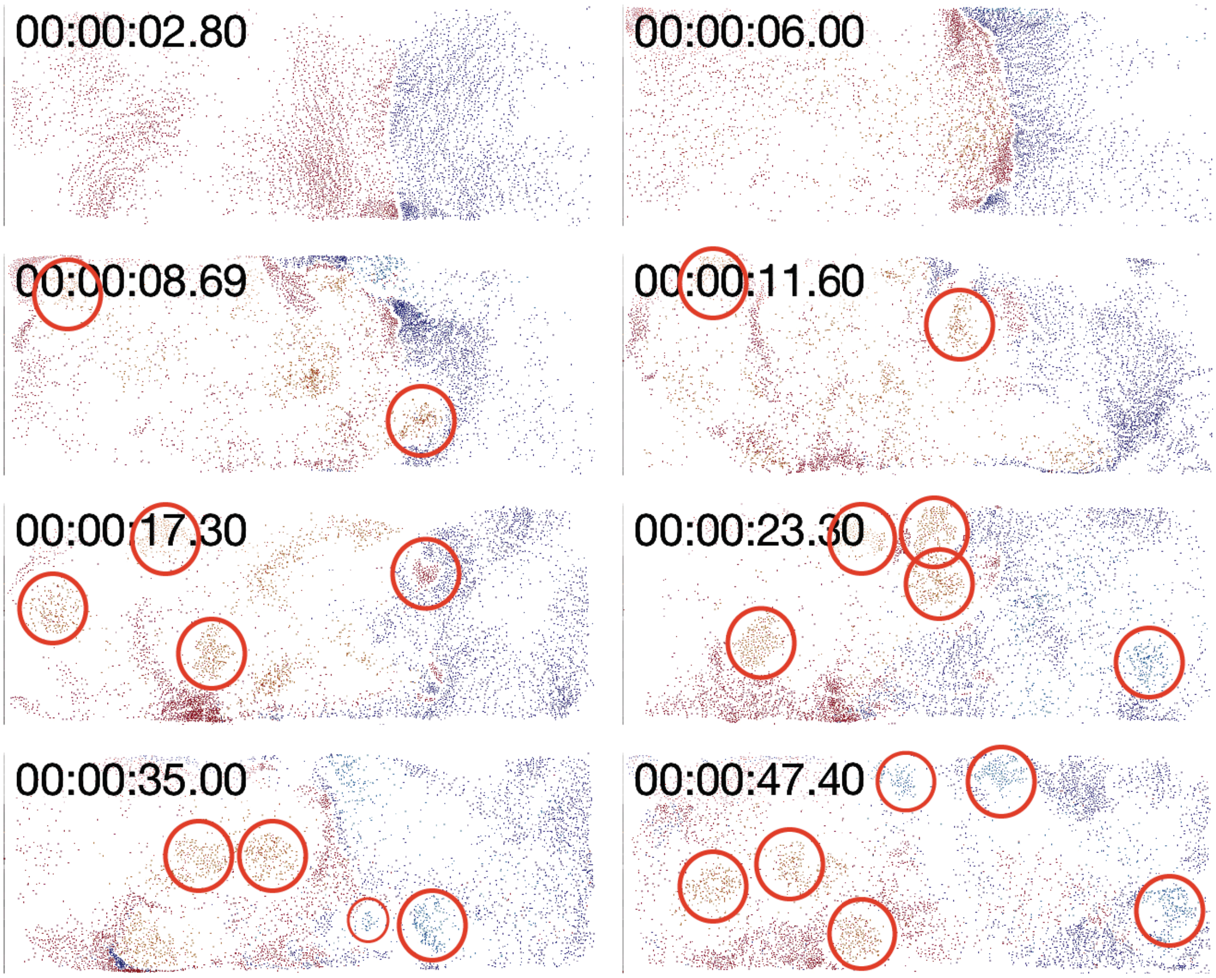

Figure 4. A sample movement of BOIDS agents with "reunion." The red circles indicate the positions where reunions are organized.

4. Reunion

As you can easily imagine, after a several minutes of mixing motion of BOIDS agents, the deformed image becomes too complicated to recognize what the original image is. Sometimes it occasionally produces beautiful dynamic visuals typically when the original camera image contains a number of different clear colors. Of course, there is no guarantee that the visitors are wearing such colorful clothes. It is more effective if the scattered elements reunite so that the visitors can recognize a part of their body is flowing. To realize this type of spontaneous reorganization, we designed and implemented reunion mechanism as described below.

The reunion is a group of BOIDS agents whose partners are located in the neighbouring place in the target area. The reorganization starts by random selection of one agent who becomes a leader of the group. For each of simulation step, each member of the group tries to find a newcomer whose partner is at the near position from its partner. This finding process is conducted targeting the BOIDS agents within the view range, together with the calculation of mutual influence in the basic collective behavior described above. When a member finds an appropriate candidate of newcomer, it sends invitation if the candidate neither belongs to nor be invited to any group of reunion. The invited agent moves toward the relatively proper position from the inviter paying less attention to the force of BOIDS behavior so that the reunion makes recognizable part of original image. The leader distributes the angle of reorganized image for each member in order for members to determine the proper relative position from the neighbouring members. A member leaves from the group when the partner lost the position in the target area. If it happens on the leader, one of the neighbouring members inherits its role. The reunion breaks when the predefined length of duration elapsed from the starting time of reuniting process. Figure 4 shows an example of movements including reunions.

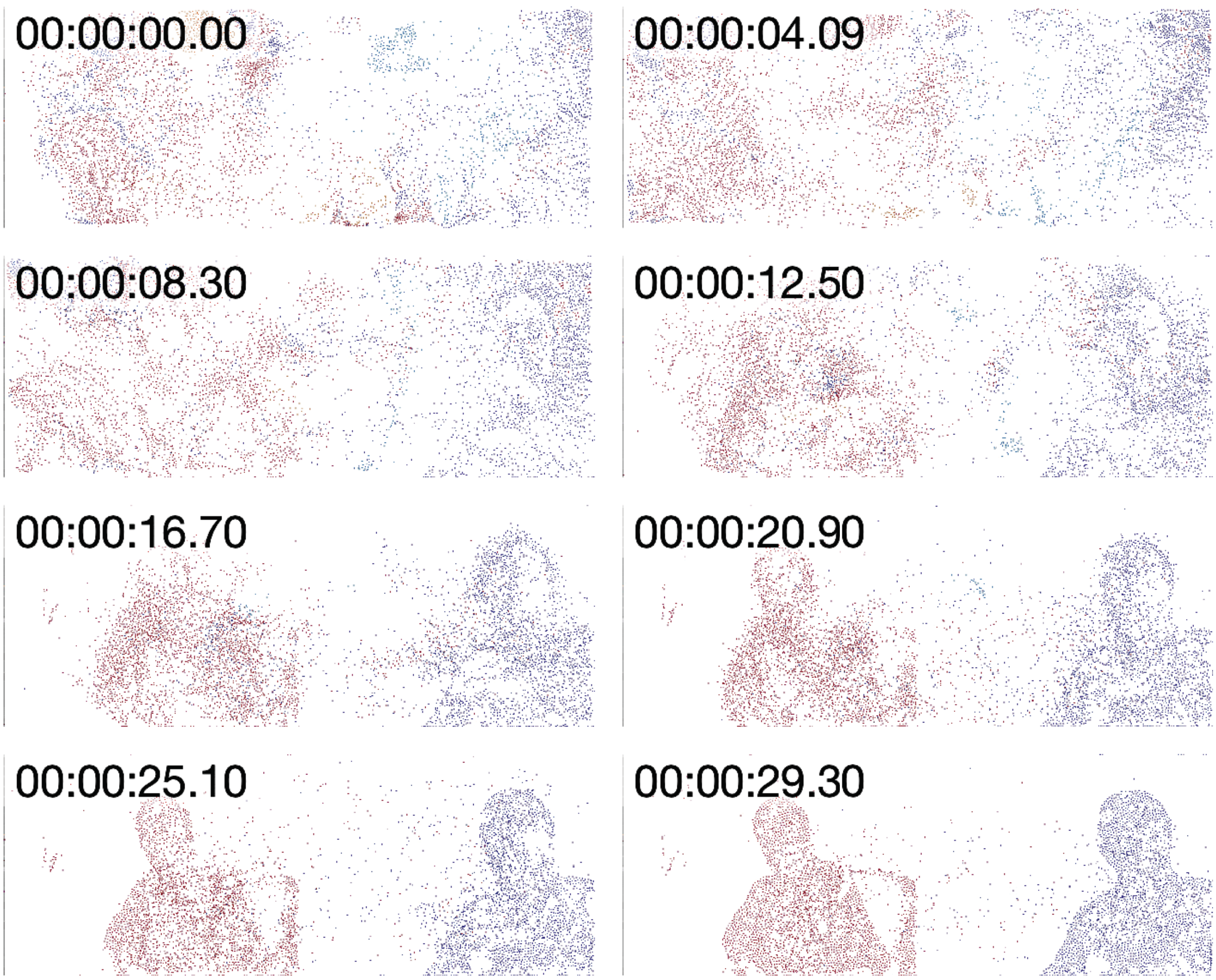

Figure 5. A sample movement of BOIDS agents on homing process.

5. Homing

We introduced a process of turning the scattered BOIDS agents back to the position of their partners for each in the final stage of appreciation by visitors. As similarly as the case of reunion described in the above section, each agent pays its attention to the partner's position as its own goal. Differently from the case of reunion, the weights of balance between flocking and homing behaviors are gradually changed as increasing the ratio of homing. Using the predefined fixed time coefficient, typically 10 seconds, the weight value of flocking behavior exponentially decreases. The agent coheres with its partner and does never move apart once it reaches the position near enough to the goal. The displayed image shifted gradually toward the camera image by dissolving transition effect after the average distance to the goal position over all of BOIDS agents became small enough. By this mechanism, the visitors' figure on the display is reformed back to the mirror image, and it makes them recognize that the show ended. Figure 5 shows an example movement of homing process.

6. Parallel processing

The system needs to compute four types of tasks in order to work as a completed interactive installation, that is, image processing, swarm simulation, image rendering, and sound synthesis. The detail of image processing and sound synthesis is described in another article [6], and the following part of this section describes the other two tasks.

The task that simulates swarm behavior is not light because it needs to care a number of agents and their mutual influences. In the installation, we use thousands of agents to obtain a complex smooth pattern that looks like not particles but liquid. To reduce the computational time, we divided the space into a lattice of 24 by 9 cells to manage the agents in order to shorten the calculation to discover the other agents in the view range for each one. The aspect ratio of grids was induced from the size of 2D space area that consists of 1,280 by 480 pixels in which two VGA camera images are horizontally arranged. This part was implemented mainly utilizing a multi-threading on multi-core CPU.

The heaviest task is the rendering process using the interpolation algorithm as described above. It theoretically requires the computational cost proportional to the multiplication of the number of pixels and the number of agents. The color value for each pixel over the rendering area is determined, referring to the coordinates map calculated with the above equation 1. It is easy to compute in parallel by storing the data in a frame buffer of GPU. To organize the map of 2D coordinates in a frame buffer from the data of swarm as the distribution of sample points, we need to accumulate the weighted 2D values for all of sample points for all of pixels. Because for each pixel the weight value of sample points at the position far away from the pixel is very small, those influences are ignorable. Instead of the iteration over all pixels, we designed the algorithm that iterates over all of sample points to accumulate the distribution of weighted 2D values onto the frame buffer within a restricted area where it affects in some degree of significance. Weighted summation is easily realized with a blending function that adds a source value to the destination value multiplied by the opacity.

Due to the improvement of GPU's power in recent years, our installation runs fast enough for smooth animation and interaction on the personal computer, such as Apple's MacBook Pro with 2.3 GHz Intel Core i7 and NVIDIA GeForce GT 750M. We are using the fixed resolution of camera image in 640 by 480 pixels for each, but the