MVA2000 IAPR Workshop on Machine Vision Applications, Nov. 28–30, 2000, The University of Tokyo, Japan

13—30 Versatile Cooperative Multiple-Object Tracking by Active Vision Agents

Norimichi UKITA, Takashi NAGAO, Takashi MATSUYAMA *
Graduate School of Informatics, Kyoto University

Abstract

In this paper, we propose a real-time multi-target tracking system by employing cooperative distributed vision system. This system consists of multiple Active Vision Agents (AVAs), a rational model of the network-connected computer with an active camera. A real-time tracking system has to dynamically change its behavior to adapt the various situation in the real-world. In this paper, we put our focus upon the dynamics of target objects. To cope with this problem, each AVA should change its target so that all the AVAs always gaze at the appropriate targets. We, therefore, designed protocols to realize a dynamic interaction scheme for cooperation among all AVAs. These protocols define the state-transition of AVA and the communication among AVAs. Using these protocols, each AVA decides its own behavior taking into account both the sharing of the observed targets' information and the other AVAs' states. As a result, all the AVAs can cooperatively track multiple moving objects as each AVA change its role and target dynamically.

1 Introduction

This paper presents a real-time cooperative distributed vision (CDV, in short) system with multiple communicating *Active Vision Agents* (AVAs, in short)[1]. AVA is a rational model of the network-connected computer with an active camera. CDV system has many advantages (e.g. wide area observation, robustness by integrating multilateral information, and flexibility of the system-organization). Facility of representing the versatile behavior is especially important because computer systems working in the real-world have to be flexible enough to adapt themselves to various and unpredicted situations.

Various vision systems can be realized by employing CDV system. Above all, a object tracking system is one of the most important basic technics for realizing applied systems (for example, autonomous surveillance and monitoring systems, ITS(Intelligent Transport System)). Hence, we have been developing a real-time tracking system by employing CDV system[1][4][5].

To realize a tracking system working in the real-world, we have to cope with various situations in the real-world that dynamically change. For example, variation of background, illumination, and unpredictable object behavior etc. Against the variation of the scene itself, a number of researches have been reported (see [2][3], for example). In this paper, therefore, we mainly addresses the problem of various and unpredicted object behavior.

In our previous works, we assumed a single target object. The system which can keep tracking multiple objects, however, is required to be able to perform a large variety of tasks. To realize a multi-target tracking system, the system has to possess the following functions.

- To grasp the information about objects in the scene (e.g. number, position).
- To cause each AVA to select an appropriate object as the target.

In this paper, we put our focus upon designing protocols to cooperate with all AVAs for multi-target tracking. These protocols define the state-transition of AVA and the communication among AVAs.

2 Cooperative Tracking System

2.1 Basic Idea for Target Tracking by AVAs

Our system consists of multiple communicating AVAs, where 1) each AVA possesses a Fixed-Viewpoint Pan-Tilt-Zoom (FV-PTZ, in short) camera, and 2) the 3D position of each camera is calibrated in advance.

The FV-PTZ camera allows us to generate background images taken with arbitrary combinations of the pan-tilt-zoom parameters from several images taken beforehand (Sec. 3 in [1]). An AVA can, therefore, detect an anomalous region by the background subtraction during widely observing the scene and adjusting the zoom. Thus the tracking by a single AVA is achievable by changing the gazing direction to the detected region in the image (Sec. 4 in [1]).

*Address: Yoshidahonmachi, Sakyo-ku, Kyoto, 606-8501 Japan. E-mail: souhaku@vision.kuee.kyoto-u.ac.jp, tnagao@vision.kuee.kyoto-u.ac.jp, tm@i.kyoto-u.ac.jp

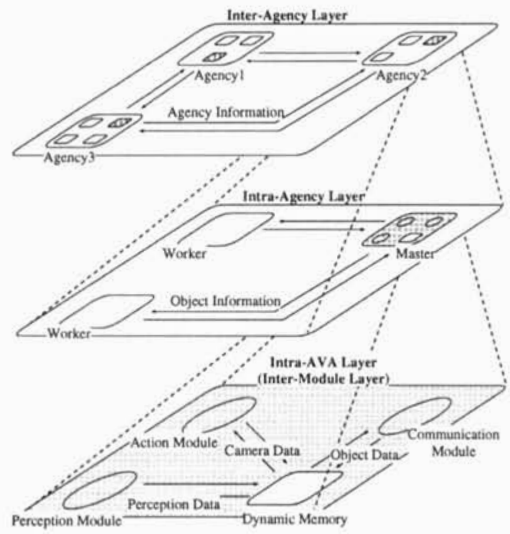

Figure 1: Three layers inside the system

All the AVAs can share their observed information through the network. For cooperative tracking (Sec. 5 in [1]), AVAs compare the 3D view lines each of which is from the camera to the detected object. If the 3D view lines can be intersected (the distance between lines is small enough), the detected objects corresponding to these view lines are considered as the same object. We can realize object identification by this procedure. Moreover, AVAs can estimate the 3D position of the object as an intersection of view lines. By obtaining the 3D position of the object, we get the following advantages.

- AVAs can keep gazing at the same object without interference of obstacles.
- AVA that has not found the object can turn to the same object by receiving its 3D position.

AVAs that cooperatively track the same object form a group. We call this group an *agency*. In one agency, only a single AVA gathers all member AVAs' detected results. We call this AVA the *master* AVA. All the other member AVAs are called the *worker* AVAs. The master AVA compares its own detected results with the other AVAs' detected results for object identification and estimation of the target's position. The history of the obtained target's information is maintained by the master AVA. If an AVA does not belong to any agency, this AVA is called the *free lancer* AVA.

Since we define an agency as the object representation in the system, events where an object appears, moves around, and disappears in the real-world are represented by the generation, continuation and elimination of agency in the system.

2.2 Layers inside System

In our system, there are the following three layers determined by type of autonomous modules that dynamically interact with each other inside each layer (Figure 1).

Intra-AVA layer This layer consists of three modules each of which have a function required for AVA. There are *perception*, *action* and *communication* modules. In addition, there is a *dynamic memory* that intermediates between three modules to dynamically share information.

Intra-Agency layer In this layer, member AVAs in one agency interacts with each other to cooperatively gaze at the same object.

Inter-Agency layer In this layer, all agencies in the system exchange information of their own agencies and targets to dynamically change the formations of agencies.

We show the detailed behavior inside each layer in the following sections.

3 Dynamic Interaction for Cooperative Tracking

3.1 Intra-AVA

In one AVA, perception, action and communication modules work independently as follows:

Perception This module continues to detect anomalous regions in the observed image. For this purpose, this module captures an observed image, and detects anomalous regions by the background subtraction.

Action Camera parameters are controlled by this module to gaze at the target or search for the target.

Communication This module exchanges detected results among other AVAs. If an AVA is the master AVA, its communication module receives detected results from the member AVAs, and sends the estimated 3D position of the target to the worker AVAs. A communication between other agencies is also carried out by the master AVA's communication module. On th other hand, if an AVA is the worker AVA, its communication module sends the detected result to the master AVA, and receives the 3D position of the target.

To work together as an AVA, each module gives other modules various information obtained by itself. Each module supplies the following information.

Perception Detected object information by the background subtraction. In our system, the 3D view lines from the cameras to each detected object are supplied as the detected results. The time when the image is captured is appended to each detected result. The action module employs, if necessary, this detected result to independently track the target. The communication module refers to this detected information for interaction between AVAs.

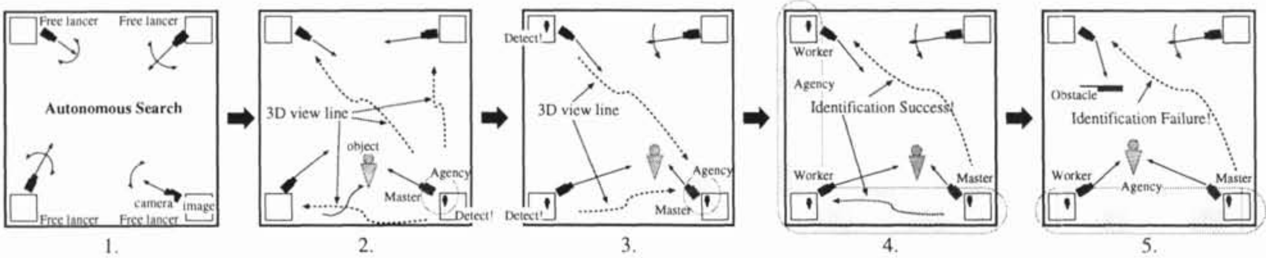

Figure 3: Agency Formation

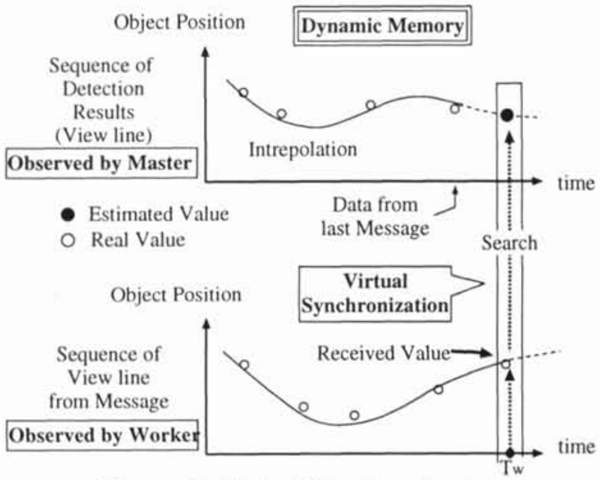

Figure 2: Virtual Synchronization

Action History of controlling camera parameters (pan, tilt and zoom parameters). The perception module needs this information to detect anomalous region in the observed image because camera parameters of when the observed image is captured is necessary for generating background images and executing the background subtraction.

Communication The 3D position of the target estimated by the master AVA. The action module refers to this information for cooperative tracking by the agency.

By separating functions of an AVA into three modules, each module can work independently. All the modules, however, have information that has to be shared among modules. If this sharing of information is realized by the direct communication between modules, autonomous dynamics of each module is disturbed by this communication. Hence, we had proposed a *dynamic memory*[4] to overcome this problem. Each module records the information into the dynamic memory, and refers to the dynamic memory whenever other module's information is necessary. Moreover, all values in the dynamic memory is recorded as time-series data. A value at an arbitrary time can, therefore, be obtained by interpolation of recorded discrete values. As mentioned above, each module can act without synchronous and sequential behavior by utilizing the dynamic memory.

3.2 Intra-Agency

3.2.1 Virtual Synchronization

In one agency, the detected results are shared for object identification and estimation of the target's position. The observed image of each AVA, however, is not synchronized because all the AVAs works independently. Object identification between the observed images taken at different time is not necessarily successful even if the detected object of each AVA is the same. To solve this problem, we employ a virtual synchronization[4] with the dynamic memory. Since the history of values is recorded in the dynamic memory, we can get the value at an arbitrary time by interpolation.

To realize the virtual synchronization, the worker AVA sends the detected view line with the time when the image is captured (T_w) to the master AVA. The master AVA, then, estimates the view line detected by the master AVA at T_w by referring to values in the dynamic memory (Figure 2). By this procedure, correct result of object identification is warranted.

3.2.2 Interaction Protocol

Inside the agency, there are three types of interactions each of which is defined as the following protocols.

Agency Formation Protocol for 'agency generation and elimination' and 'joining into the agency and exit from the agency'.

At first, all AVAs search around for an object autonomously as free lancer AVAs (Figure 3, 1.). A free lancer AVA, that first finds the target, generates an agency and becomes the master AVA (Figure 3, 2.). If an agency exists when a free lancer AVA detects an object, this free lancer AVA requests object identification from the master AVA of the agency (Figure 3, 3.). If this object identification is successful, this free lancer AVA joins into this agency as a worker AVA (Figure 3, 4.). Otherwise, this free lancer AVA generates a new agency. A worker AVA, that loses track of the target, leaves the agency and returns to the free lancer AVA (Figure 3, 5.). When all AVAs fail in tracking the target, the agency disappears.

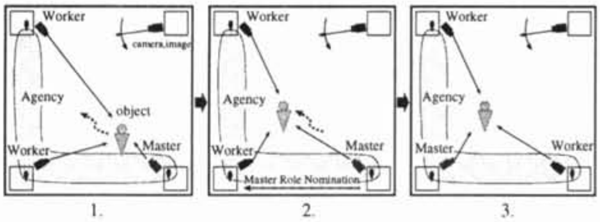

Figure 4: Role Assignment

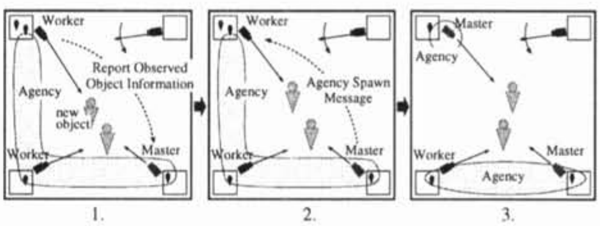

Figure 5: Agency Spawning

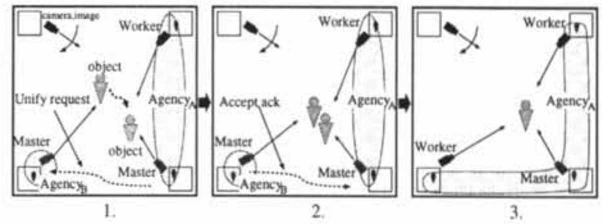

Figure 6: Agency Unification

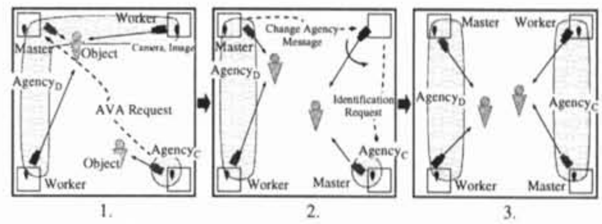

Figure 7: Agency Restructuring

Role Assignment Protocol for stable continuation of the agency.

Since a target object moves in the real-world, an AVA that first becomes the master AVA will not necessarily always gaze at the target. Moreover, the reliability of the master's detected result[1] is always crucial for estimation of the target's information. While tracking, therefore, the master AVA dynamically changes inside the agency so that the master AVA always has the most reliable object information (Figure 4, 2. 3.).

Agency Spawning Protocol for new agency generation from the agency.

When the member AVA of the agency finds a new object, this AVA generates a new agency for tracking this new object. If the master AVA fails object identification as a result of comparing the history of the target information with received detection (Figure 5, 1.), this detected region is regarded as a new object. The master AVA, then, orders the member AVA that finds the new object to generate a new agency and to become a master AVA of the new agency (Figure 5, 2.). Finally, the new agency is generated (Figure 5, 3.).

3.3 Inter-Agency

In our system, each agency keeps tracking its target object. All agencies have to exchange their member AVAs with each other according to the movements of the targets and conditions of other agencies. For this purpose, all the agencies send mutually messages that have 1) the target's information (3D position), 2) the agency's information (entry of member AVAs). When the master AVA receives a message from another agency, this master AVA executes two functions, object identification and comparison between mutual member AVAs' entries.

3.3.1 Interaction Protocol

To effectively assign all AVAs to agencies, we introduce the following two protocols.

Agency Unification Protocol for unifying agencies whose targets are considered as the same object as a result of object identification between agencies.

Although detected objects are first regarded as different objects in the real-world, it is conceivable that these objects will be considered to be the same in subsequent object identification. For example, such an event happens in the following cases.

- When different objects become close enough (shown in Figure 6).
- A single object is regarded as two objects due to the failure of object identification.

The master AVA of $Agency_A$, that regards the target of another agency ($Agency_B$) as the same as the target of $Agency_A$, asks the master AVA of $Agency_B$ to join (Figure 6, 1.). The master AVA of $Agency_B$, then, replies with a message of acceptance (Figure 6, 2.), and joins into the negotiating agency. At this time, the master AVA of $Agency_B$ orders all the member AVAs of $Agency_B$ to join $Agency_A$.

Agency Restructuring Protocol for exchanging member AVAs among agencies so that abilities to track the target of all agencies are uniform.

By comparing entries of member AVAs, The master AVA of $Agency_C$ decides to demand member AVAs from another agency ($Agency_D$) in the following cases.

[1] The reliability is determined by 1) the size of the detected region in the image, 2) the distance from the object to the camera in the scene 3) time when observed image is captured, and so on. The weight of each factor is established according to the given task for the system. In addition, the observable-area model[5] is useful for a determination of the reliability.

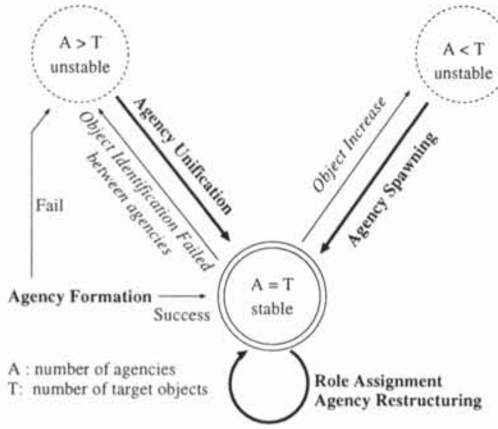

Figure 8: State Transition of System

- The number of each member AVAs is much different.
- An agency has a member AVA that is suitable for tracking a target of another agency[2].

In the above instances, the master AVA of $Agency_C$ requests member AVAs from the master AVA of $Agency_D$ (Figure 7, 1.). The master AVA of $Agency_D$, then, elects the AVA that should transfer to $Agency_C$. If $Agency_D$ can keep tracking the target in the stability though the member AVA leaves, the master AVA orders the elected AVA to join $Agency_C$ (Figure 7, 2.). Finally, The AVA that is ordered to be transfered joins $Agency_C$ (Figure 7, 3.).

4 State Transition for Cooperative Tracking

By various interaction protocols mentioned above, the system tracks multi-objects as the system changes its own formation. Figure 8 shows a state-transition of the system. When 1) the system makes a mistake and 2) the situation in the real-world changes, the system returns to the stable state by causing the dynamic interaction between AVAs.

The changing of the AVAs' behavior achieves the above state-transition of the system. Figure 9 shows an epitomized state-transition model of AVA. Each frame and arrow in Figure 9 means respectively a state and a state-transition of AVA. The state-transition of AVA is caused by the following factors.

- Received message (shown by a **bold** font in Figure 9).
- Detected and estimated results of the target information (shown by a *slanted* font in Figure 9).

In accordance with the behavior of the system and AVA mentioned above, the system cooperatively keep tracking the target in the dynamic situations.

[2]This aptitude is also determined by the same criterion of the reliability of AVA's detected result.

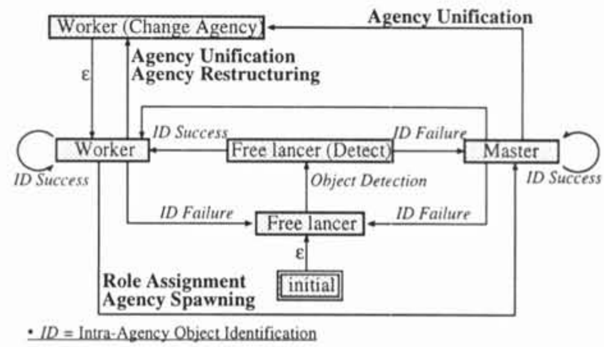

Figure 9: State Transition of AVA

5 Concluding Remarks

We proposed a general idea for multiple-object tracking system by employing cooperative distributed vision system. In this system, all AVAs autonomously and asynchronously interact for cooperation among AVAs.

In this paper, we focused upon the dynamic interaction of AVAs and designing a protocols for this interaction. To realize a versatile tracking system that can control its behavior for the given task (e.g. surveillance of wide area, navigation of a specified person, etc.), however, the system should be adaptable for task specification. We are, therefore, designing functions for such a versatile tracking system, and developing a prototype system to verify the effectiveness of our proposed system.

This work was supported by the Research for the Future Program of the Japan Society for the Promotion of Science (JSPS-RFTF96P00501).

References

[1] T. Matsuyama, "Cooperative Distributed Vision - Dynamic Integration of Visual Perception, Action and Communication -", Proc. of Image Understanding Workshop, pp. 365–384, 1998.

[2] C. Stauffer, E. Grimson, "Adaptive background mixture models for real-time tracking", Proc. of CVPR99, Vol.II, pp.246–252, 1999.

[3] I. Haritaoglu, D. Harwood, L. S. Davis, "A Fast Background Scene Modeling and Maintenance for Outdoor Surveillance", Proc. of ICPR2000, Vol.4, pp.179–183, 2000.

[4] T. Matsuyama, S. Hiura, T. Wada, K. Murase, A. Yoshioka, "Dynamic Memory: Architecture for Real Time Integration of Visual Perception, Camera Action, and Network Communication", Proc. of CVPR2000, pp.728–735, 2000.

[5] N. Ukita, T. Matsuyama, "Incremental Observable-Area Modeling for Cooperative Tracking", Proc. of 15th ICPR, Vol.4, pp.192–196, 2000.