INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29/WG11
MPEG97/M3399
Tokyo 16-20 March, 1998

Title: **Visually Based Control: another Application for MPEG-7**

Author: **Stephen PALM, Taketoshi MORI, Tomomasa SATO**
Contact: **Stephen Palm T: +81-3-5606-7169 F: +81-3-5606-7259**
E: palm@lssl.rcast.u-tokyo.ac.jp University of Tokyo, RCAST

In reviewing the current MPEG-7 Applications version 3(W2084), it appears that there is one other category of applications that could be considered for inclusion. In the field of robotics, there have been several developments in the area of visually based control. Instead of using text based approaches for control programming, images and image sequences are used to specify the control behavior and are an integral part of the control loop (e.g. visual servoing). This new MPEG-7 application could be considered one of the "Professional Applications" under a section entitled "teleoperation and robotic visually based control".

One of the aspects of the control is the description of control information between (video) objects that are not necessarily associated via temporal spatial relationships. Contribution M2878 discussed one approach for description of arbitrary associations. We wish to encourage work in this area.

The attached draft article describes some work-in-progress to develop visually based control methods in the context of MPEG-4 using some preliminary concepts from MPEG-7. Pages 6-7 and then pages 11-12 (Behavior Sampling for Cell Handling) cover the specific discussion of the use and need for the "control scene description" in addition to the spatial temporal scene description. Currently, they are referred to as Node Control Associations (NCA).

Brief description of our overall work:

> We have developed the *bilateral behavior media* (BBM) paradigm based upon explicit visual communication between human operators and their teleoperated tools. The bilateral behavior media paradigm comprises three areas: 1) A control methodology for visually specifying tasks and visually controlling machines. The methodology is referred to as *status driven* control. 2) A data representation and extraction method for accumulation of visually based interactions between humans and tools. This is termed *behavior sampling*. 3) Functions for assisting and supporting humans through visual mechanisms. Capabilities include visually navigated “redo” or “undo” based upon past visual-control sequences. This functionality is expressed as *status on demand.* Together the three areas encompass the notion of bilateral expression of behavior between humans and machines through a multiplicity of visual media.

For non-spatial temporal scene descriptions it seemed that AAVS might be appropriate. However, it seems that AAVS is more suited for the modification of existing scene descriptions. Of course, it does seem that AAVS might be useful in the extensive User interaction required in teleoperation, however it seems that an enhanced scene description is also requisite.

Bilateral Behavior Media: Visually Based Teleoperation Control with Accumulation and Support

Visually based control, accumulation and support systems are presented as an effective environment for teleoperation. We developed the Bilateral Behavior Media (BBM) paradigm and implemented systems which offer a status driven interface, collection of sampled control behavior, and operator assistance. The paradigm emphasizes a human-intuitive visual specification of task completion states without resorting to image understanding, modeling or extensive calibration. Visually based teleoperation has been effectively applied in a variety of microworld tasks where an operator's past experience in the macroworld is not applicable to the physics and scenario experienced in the microworld. Experiments of manipulating individual biological cells and microworld assembly have shown the success of the visually based BBM paradigm.

Keywords: Visually based teleoperation, microworld manipulation, accumulation

Stephen PALM Taketoshi MORI Tomomasa SATO
RCAST, The University of Tokyo
4-6-1, Komaba, Meguro-ku, Tokyo 153-8904, JAPAN

Teleoperated robots are indispensable in environments where humans cannot perform direct manipulation. In dangerous or distant locations or in the microworld, humans must manipulate the environment through a remotely controlled mechanism. Although teleoperation techniques have been extensively researched and developed, human operators still experience problems in accomplishing tasks when working through machines. Improving the teleoperating worker's situation is our underlying theme.

Recent work in visually based control methods has laid the foundation for advanced and robust master slave teleoperation. The visually based control methods offer 1) a more intuitive human machine interface and 2) allow for much simpler and economical control algorithms. [1, 2, 3, 4]

Previous generation teleoperation control methods such as "joint-angle correspondence" or "generalized coordinate transform" techniques can offer very good response times or be used with automated control techniques. However, there are also several significant limitations. For example, joint-angle techniques require that the master and slave arms must be of identical mechanical configurations or autonomous control by the robot is not possible. Likewise, for generalized coordinate transform techniques, inclusion of external sensing information such as video or adaptation to dynamic environments has proven to be difficult.

With many control techniques, appending sensors, especially visual sensors such as cameras, has been attempted in order to improve the human's understanding and control of the remote environment. However, we contend that the control method should be fundamentally based on sensing and in particular visual sensing in order to be effective in real world teleoperation applications.

We have developed the *bilateral behavior media* (BBM) paradigm based upon explicit visual communication between human operators and their teleoperated tools. The bilateral behavior media paradigm comprises three areas: 1) A control methodology for visually specifying tasks and visually controlling machines. The methodology is referred to as *status driven* control. 2) A data representation and extraction method for accumulation of visually based interactions between humans and tools. This is termed *behavior sampling*. 3) Functions for assisting and supporting humans through visual mechanisms. Capabilities include visually navigated “redo” or “undo” based upon past visual-control sequences. This functionality is expressed as *status on demand.* Together the three areas encompass the notion of bilateral expression of behavior between humans and machines through a multiplicity of visual media.

In this article we describe the implementation of the status driven microhandling system (SD-MHS) and a behavior sampling system combined with a cell handling system (BS-CHS). Although applicable in any teleoperation domain, we have concentrated our application of BBM techniques to teleoperation in the microworld. The microworld offers a challenging teleoperation environment because many of the experiences of physical property manifestations that humans have acquired in the macroworld are not applicable in the microworld. For example, the magnitude of adhesive forces is relatively much stronger than gravitational forces in the microworld. Objects are more likely to "stick" to the manipulator than to "fall" due to gravity.

Another application of BBM techniques is the biological world. Recent studies of aging and high fat diets have focused on analyzing Mato fluorescent granular perithelial (FGP) cells [5]. New techniques to analyze individual cells require the isolation of each cell by removing the tissue surrounding the cell. Figure 1 shows a visually based manipulator with a two micrometer wide scraper made of glass. The manipulator scrapes the undesired tissue from around the Mato FGP cell in preparation for its removal.

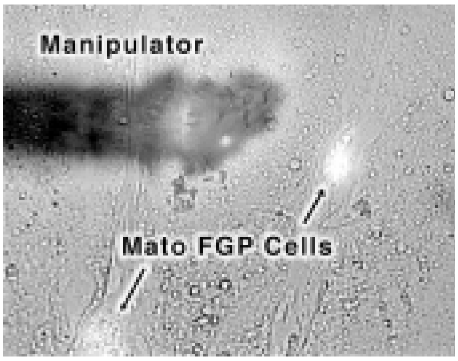


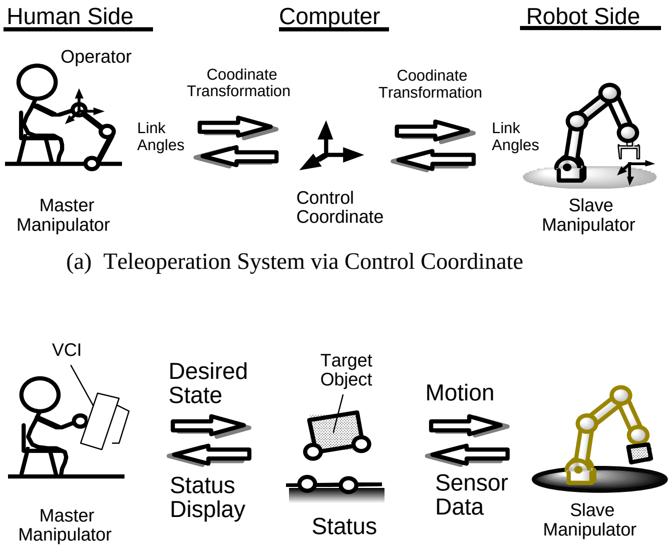
Figure 1. Cell manipulation environment.

We will first present and discuss the three main components of the BBM paradigm: status driven control, behavior media, and status on demand. This will be followed by a discussion of the implementation of the systems that perform status driven control and behavior sampling. Finally, we review the experiments performed to show the productivity of using BBM techniques.

THE BILATERAL BEHAVIOR MEDIA PARADIGM

Status Driven Control Method

The main components and information flow of the status driven master slave control method are illustrated in **Figure 2**(b). The operator interfaces with a computer screen (labeled VCI) instead of a traditional master manipulator. The slave manipulator itself can be any conventional manipulation arm. The implementation of a status driven control method requires and produces different information than conventional teleoperation control methods (e.g. **Figure 2**(a)). In particular, data from sensors in the slave environment is essential to the system control and the task specification by the operator.



(a) Teleoperation System via Control Coordinate

(b) Teleoperation System via Status

Figure 2. Teleoperation System Comparison

The status driven control method relies on sensor information from the slave environment. The integration of sensor information into the control algorithm is accomplished through *sensing points*. Sensing points are used to describe pertinent features in the workspace. *Target sensing points* are associated with the target object that is to be manipulated. Once designated, the target sensing points remain fixed to the object as the object is moved throughout the task. Likewise, environmental sensing points are associated with pertinent features in the manipulation environment. *Environmental sensing points* are similar to target sensing points in that they also remained fixed to the environmental location they were assigned. If the environment moves with respect to the viewing frame of reference, the sensing points would be made to track that movement. Environmental points differ from target sensing points in that the environmental sensing points are also the goal states to which the target sensing points are being made to coincide during automatic mode. In other words, the system controls the manipulator is such a way as to cause the target sensing point(s) and environment sensing point(s) to coincide.

In general, the occurrence of sensing points in a status driven task can be divided into two phases: task teaching and task execution. Control of a target object is initiated by associating sensing points to it. This process is referred to as the teaching phase and is generally only performed once in a control situation at the beginning of the session. The operator interfaces with a video image of the work environment where the sensing points will be superimposed as they are being selected. The work environment image, status and query text, the sensing point input method, and superimposed symbolic display constitute the so-called visual communication interface (VCI).

After the task has been described to the system during the teaching phase, the task execution phase begins. In automatic mode, the system attempts to make the corresponding target and environment sensing points coincide. In operator and system shared mode, the system causes the target sensing point to follow the operator's relocation of environment points.

We illustrate the process with a 2D example of placing on object on another object. Consider a placement example in two dimensional space where the system completely controls the motion after the teaching phase. The teaching phase requires specifying the type of task and the pertinent attributes of the target object and slave environment by specifying sensing points. Two sensing points would be designated with respect to the target in the image to indicate the object's edge that will be placed on the environment surface. Two additional sensing points would be designated with respect to the work environment to indicate the surface on to which objects may be placed or an area where objects must not be allowed to enter. (See the circles in Figure 7)

After the teaching phase, the system automatically controls the manipulator. As the target begins to move, the visual sensors track the movement so that the target's sensing points remains fixed with respect to the target object. Movement continues until the target sensing points coincide with the environment sensing points. Once they coincide, the movement stops since the task has been completed.

One particular point of the status driven teleoperation input system is that instructions of movement to the slave are operations on the sensing points. The manipulator itself is not being controlled by the master, instead the slave component of the system moves the manipulator in response to the desired status of the sensing points as described by the master. Movement of the manipulator is caused by the relative locations of the sensing points.

Behavior Sampling Method

The status driven control method described so far is basically a very short term control aspect. Status driven control is concerned with the immediate task of causing the sensing point to coincide to achieve task completion. However, use of tools is much more than a disjoint set of independent events. For long term considerations we believe the following features are useful.

1) A data representation and storage mechanism is needed for visually based control mechanisms. The system can only display an instantaneous representation of the control sequence. The heretofore described status driven system does not provide a means for displaying past control sequences. The only method of reviewing a past manipulation would be to video tape the combined video image and graphical overlay of the control indications.

2) A means to accumulate the underlying raw data (both object visual representation and control instructions) is needed. While some knowledge based or autonomous robots do have mechanisms for accumulating past experience, they tend to be based purely on image frame data or abstract representations or models.

3) The ability to syntactically organize the visual and control information and allow for the addition of semantic information. A status driven system purposely does not have an underlying concept of the objects it is manipulating. However, when humans are reviewing the information it would be useful for the operator to annotate some of the criteria and information that they used in selection and placement of the sensing points.

The input to a behavior sampling system consists of the video image of the slave environment and the time stamped control information. The control information includes such items as the location of the objects in the environment and the type of control desired. The control information is typically specified by sensing points and the desired relationship of the final state of the sensing points. All of this information will be processed and converted into the behavior sampling data representation. Further, if an operator wishes to input semantic information for a given node or link, the data representation is capable of annotating such information to the nodes and links. Semantic annotation is possible both in the initial creation phase and in post-creation phases.

The output of behavior sampling is an indexable, structured stream that contains both the visual and control information. The form of the stream is such that addressing of individual objects or information is readily obtainable without resorting to decoding all of the information in the stream or even in a large segment of the stream. The stream is suitable for storage (e.g., hard disk) or for transmission. Further, the output is parsable in such a manner that the form and style of the control performed on a given object in a past sequence is usable for control of a different object in a future situation. In other words, the behavior sampling output is suitable as the input to the status on demand functions.

Behavior Sampling is partially based on the concepts of hypermedia and syntactical or semiotic analysis. In the visual domain, a basic element is the designation of visual representations of the objects in the environment and their spatial and temporal locations in the scene. These are referred to as *visual objects.*

In addition to visual objects, there are several other channels of information in a visually based control system. The control information and relationships specified by the sensing points as well as the graphical and textual system display information must be communicated between the teleoperator and the system.

The syntactical or semiotic analysis method exploits the underlying structure of the real world scene in the representation. Syntactic methods extract structural information without understanding the meaning or semantics of the visual objects since the elements can be derived through low-level vision techniques. The structure of the visual and control information is extracted by observing *signs*. The first two columns of Table 1 show Gonzalez's proposed assignment of signs for the images and video domains [6]. The third column shows the control domain signs developed specifically for behavior sampling.

Table 1. Assignment of signs for various media domains.

	DOMAIN		
	Images (Spatial)	Video (Temporal)	Control (Visual)
Meta Sign	Picture	Episode	Completed Work / assembly
Signs	Objects	Scene	individual object task
SubSigns 1	Surfaces	Shot / Global Motion	sensing pairs
SubSigns 2	Lines	Objects / Local Motion	sensing points
SubSigns 3	Pixels	Stationary Change	

Typically the extracted information is structured into a hierarchical tree-like structure. Traditional scene description techniques are able to describe the spatial temporal relationship between objects and the association of a sensing point to an object. However, a scene description alone is inappropriate to describe the overall change between the objects (as described by the control relationships between sensing point pairs). Each sensing point of the pair would be part of separate objects in the scene description and therefore would not have direct links in the description tree.

We introduce an enhanced tree structure with *node control associations* (NCA). NCA allows explicit linkage of control information directly between arbitrary nodes through "associative information" in a higher common node. (See Figure 3) These associations can be arbitrarily added and deleted to indicate the control information changes between nodes. The most common case is the association of control information between an individual sensing point in one visual object with an individual sensing point in another visual object. In total, NCA describes three types of information in the nodes: 1) *generic*: information applicable to the node and all its descendants; 2) *intrinsic*: information applicable only to the node and not its descendants; and 3) *associative*: information that relates two or more of the node's descendants [7].

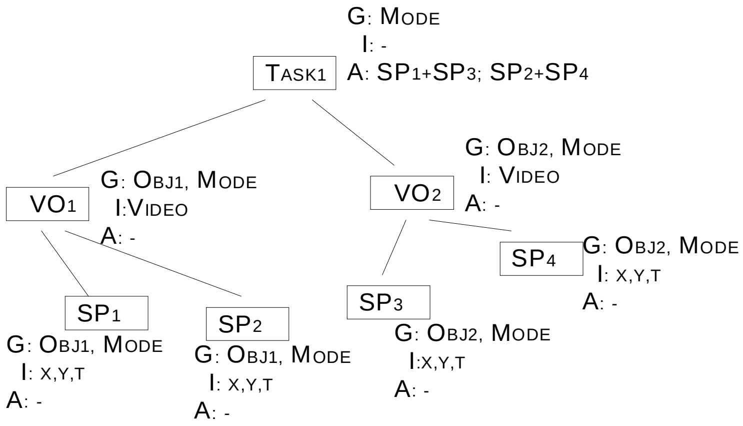


Figure 3. Visual and Control Tree. Task, Video Object (VOx), and Sensing Point (SPx) spatial information is represented in linked boxes. The adjacent Generic (G), Intrinsic (I), and Associative (A) NCA fields describe the control information.

The techniques for syntactical analysis are based upon observing the signs listed in Table 1. Syntactical analysis occurs for both the visual and control data domains. Extracted information from the two domains is complementary in that observation of a sign in one domain is often useful for structure segmentation in the other domain.

Segmentation of automated control actions is rather simplistic since the control actions occur in well-defined states and relationships. For manual operation by the operator, temporal changes are evaluated. Both the operator input events and the manipulator control system status feedback are analyzed. Events that are extracted include definition of new status points, when status points coincide, relationships between status point sets, segues, and task transitions.

The lowest level of the visual structuring begins with statistical analysis of the image sequence. Detected signs include motion of the visual objects, coincidence of visual object surface projections, global motion, and other significant changes in image statistics. Monitoring of global geometric translations and rotations allows indirect monitoring of camera work. For example, global motion can indicate a change in the operators intended work area. Likewise, analysis of localized translations and rotations indicates manipulation of the visual objects.

During the recording phase, the preservation of all aspects of the video is typically redundant for capturing the essence of the control state changes. Although the control system needs to monitor an image tracking window surrounding each sensing point, the control system does not need to record the entire visual object. However, for later analysis by the status on demand functions, the system records the individual visual objects. Also, the recording of the visual objects is handy for human observation of the progress. Often the images of the manipulated objects themselves are sufficient for human understanding of the manipulation. The “background” both literally (in the image) and figuratively (the objects not represented by sensing points) is often unnecessary for human understanding. Thus the actual amount of imagery that is recorded and presented is often spatially and temporally sub-sampled based on its relevance to the motion behavior of the operator interacting with the manipulator. However the background information should not be completely eliminated. In some cases, humans may expect to see some type of background as there always is “background” in any natural scene, however the updating of imagery of the background can often occur only initially or rather infrequently.

Although the extraction of behavior from the images and status points is syntactically based, the data representation also allows semantic information to be annotated. Some semantic information can be automatically generated after syntactical analysis and using a knowledge base in manipulation situations with a priori information. Semantic information can be inferred from such sources as assumptions about the tool being used and the functions it can perform; assumptions about the objects and their roles; and operator labeling of context. When encoding control information, behavior sampling considers the inputs of the operator, changes in status points, as well as changes in the actual image. The syntactic aspect of behavior sampling itself does not directly infer the operator's task intention, however semantic information can easily be by associated/annotated with the syntactic stream.

Status on Demand

The status on demand functionality is a visually based interface to the behavior sampled data in a status driven system. The status on demand system is able to display, through imagery, graphics, and text, milestones in which (task) status has transitioned from one type of task to another. Thus, an operator is able to view the past sequence of events comprising tasks in an easy to comprehend and partition manner. The task status at each relevant point in time of the procedure is then available for reference and visual re-manipulation by the operator.

A status on demand system can have several levels of functionality. Lower level functions provide immediate short term support for the operator to modify a recent manipulation. Intermediate level functions including editing of manipulation parameters or annotation of semantic information. Higher level functions would allow the replay or reuse of previous manipulation procedures in new control situations. We describe a low level function that has already been implemented to verify the behavior sampling functionality.

Redo allows the operator to repeat the last style of change of state (perhaps on a different set of sensing points). This is useful with similar motions that need to be performed multiple times from (typically) different start and end points. For example, if there is a series of objects to be manipulated in a similar way, the operator would setup the sensing points for the first object and perform one or more manipulation tasks on it. For the subsequent objects, new sensing points would be used to specify the object(s) and the Redo function would perform the same compound set of manipulations.

SYSTEMS DESCRIPTION

In this section we describe the status driven microhandling system (SD-MHS) and a behavior sampling system combined with a cell handling system (BS-CHS). The status on demand system is currently under development, however some status on demand functions have already been implemented to test the functionality of the behavior sampling system.

Status Driven Microhandling System

The first implementation of a status driven teleoperation system is the status driven micro handling system (SD-MHS). This system provides the operator several methods for manipulating objects on a micrometer scale. The hardware, software, and interface will be described in this subsection.

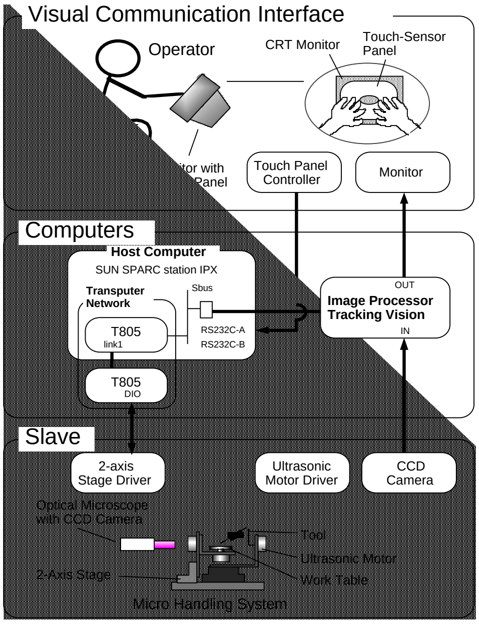


Figure 4. Status Driven MHS Block Diagram

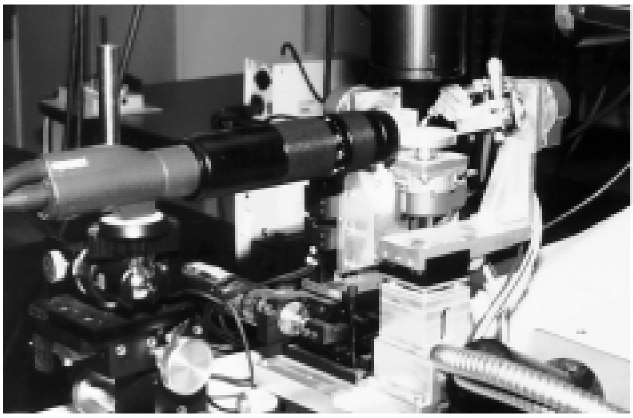
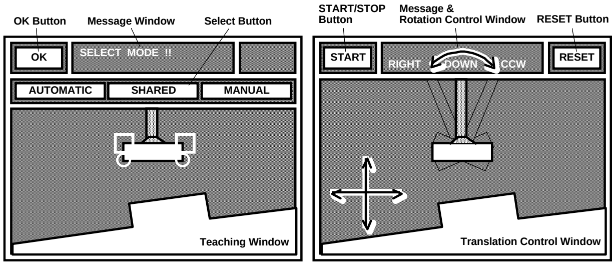


Figure 5. Micro Handling System I (MHS-1)

A schematic diagram of the overall system is shown in Figure 4. The slave portion utilizes the Micro Handling System I (MHS-I) described in [8] and shown in Figure 5. For this system implementation, MHS-I views and controls the slave environment in two dimensions with three degrees of freedom through a two-axis work table with 17.4nm resolution and a tool that can be rotated with 0.1° resolution. The operator communicates to the master through the hardware components of the VCI which consist of a video monitor with an integrated touch panel. The video monitor continuously displays a view of the slave work area obtained from an optical microscope and CCD camera mounted close to the work area. System command and control software is implemented in C and LISP running on a Sun workstation, two Transputers, and Fujitsu image tracking hardware.

When operating the system, the operator teaches the sensing points and can direct the slave milestone motion by appropriate finger movements on the touch screen. In addition to the image of the slave work environment, computer generated symbols, text, and graphics are overlaid on the monitor. They are used to prompt the operator during the teaching phase and symbolically display the location and movement of the sensing points as shown in Figure 6.



(a) Teaching Mode (b) Execution Mode

Figure 6. Visual Communication Interface of SD-MHS

The location of the sensing points should be selected by the operator based on two criteria. First, each sensing point should be placed on a relevant physical attribute of the target or environment objects. For example, sensing points could be placed on edges or vertices. Second, since the system manipulations will eventually cause the target sensing points and the environment sensing points to coincide, the sensing points should be located such that coincidence would indicate the final desired state of the target with respect to the environment.

After an object or environment sensing point is specified by the operator, the system must maintain the sensing point in the same relative position on the physical element even though it is moving with respect to the video camera frame of reference. In this system implementation, that sensing point synchronization is accomplished through using vision tracking hardware. The tracking hardware uses template matching techniques to continuously update the location of the center in each of the several tracking windows in the image. The operator selects the initial center of each tracking window. Geometric methods are then used to keep the sensing points in the appropriate location with respect to the tracking window. Examples of typical placement of sensing points and tracking windows centers in a microworld manipulation are shown in Figure 7.

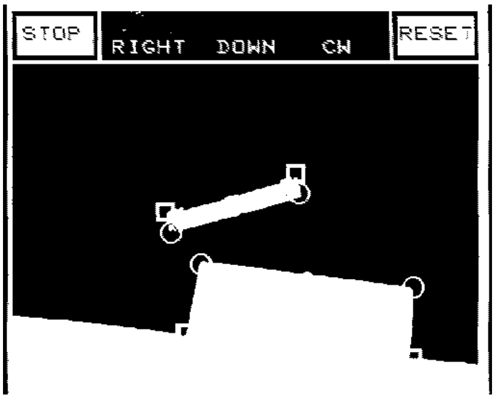
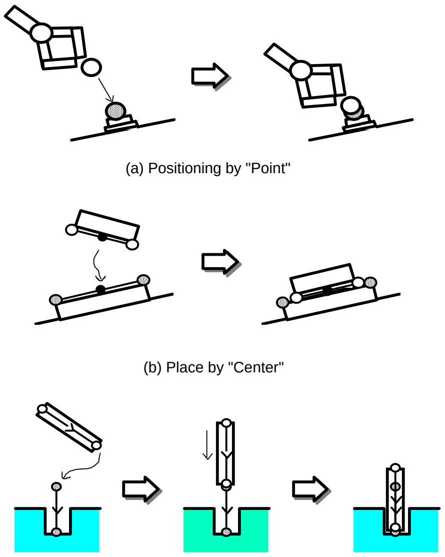


Figure 7. Example of object and environment sensing points (indicated by circles) and the corresponding tracking windows (window centers are represented by squares).

Although the use of status driven techniques are applicable to a wide range of assembly tasks, in this system we concentrate on a few exemplary tasks and the functions necessary to execute them. The functions correspond to the relationship between the initial and final locations of the sensing points. When teaching the sensing points to the system, the operator selects one of the task functions so that system knows the desired final state of the sensing points. Different task functions require differing numbers of sensing points. The functions defined here should not be considered exhaustive or intrinsic. For more complicated assembly tasks, these functions can be combined or new functions could also be implemented.

In the fully automatic mode, three task functions have been implemented: “point”, “center”, and “arrow” which progressively have more constraints on the movement from initial to final state. These three functions, superimposed on tasks in the slave world, are illustrated in Figure 8 and described as follows.

The “point” function operates on two points, moving one point so that it coincides with another point. In general, the initial position of the moving point would represent a sensing point on a target object in it’s initial state. This function can be executed with purely translational movement. (See Figure 8(a)).



(a) Positioning by "Point"

(b) Place by "Center"

(c) Insert by "Arrow"

Figure 8. Automatic Mode Task Functions. Sensing point are represented by the circles.

The “center” function is similar to the “point” function but operates on two pairs of sensing points. One pair of points is associated with the target and the other pair is associated with the environment. Each of the two points on an object forms a line segment that can be thought of as a surface in the slave world. In order for the target and environment points to coincide, the two line segments must be also be coincident. Manipulating one of the lines to be parallel with the other may involve rotational movement in conjunction with the translational movement to make the points coincident. This function actually uses 6 points (see Figure 8(b)), four explicit and two implicit points. The explicit points (white or gray circles) are the end points of the line segments. The implicit points (black circles) are the midway locations on the segments which are actually made to be centered. The function is applicable for placement tasks.

The “arrow” function also operates on two line segments and utilizes both rotational and translation movement. In this function, the two segments are also made to be coincident, but one set of endpoints (the “heads” of the “arrows”) is required to be coincident instead of the midpoints. (See Figure 8(c).) Two additional constraints in the movement control are: 1) the head of the moving arrow will first be made coincident with the tail of the stationary arrow and 2) then the arrow axes will be made collinear before the final translational movement along both arrow’s axes.

Rotational and translation motions each have two speeds which we term “Fast” and “Slow”. Selection of the translation speed is dependent on the distance between the sensing points. Selection of the rotation speed is based on the angle between the pairs of sensing points. The movement thresholds are shown in Table 2 and Table 3. The movements are executed five to ten times per second depending on processing time. The movements are assumed to be slow enough to ignore dynamic effects.

Table 2. Translational Movement Thresholds

	step output (pixels)	**difference** (pixels)
Stop	0	$0 \le x < 2$
Slow	1	$2 \le x < 20$
Fast	3	$20 \le x$

Table 3. Rotational Movement Thresholds

	step output (degrees)	**difference** (degrees)
Stop	0	$0 \le \theta <3$
Slow	2	$3 \le \theta < 10$
Fast	5	$10 \le \theta$

Behavior Sampling for Cell Handling

The recording and display composer mechanisms of the first behavior sampling system are based upon the draft MPEG-4 framework [9]. MPEG-4 provides a toolbox of functions for video encoding such as specifying and encoding individual objects and specifying how the individual objects are composed to form a complete scene. MPEG-4 does not provide mechanisms for segregating or extracting objects from a video frame nor does it provide a mechanism for describing control relationships between objects.

Implementation of a behavior sampling system entailed developing two main components 1) an automated video segmentation method and 2) control information processing and storing. These are shown in the dashed boxed in Figure 9. The manipulator control section is similar in function to the SD-MHS control system. The object and scene encoding and decoding functions are part of the MPEG-4 framework.

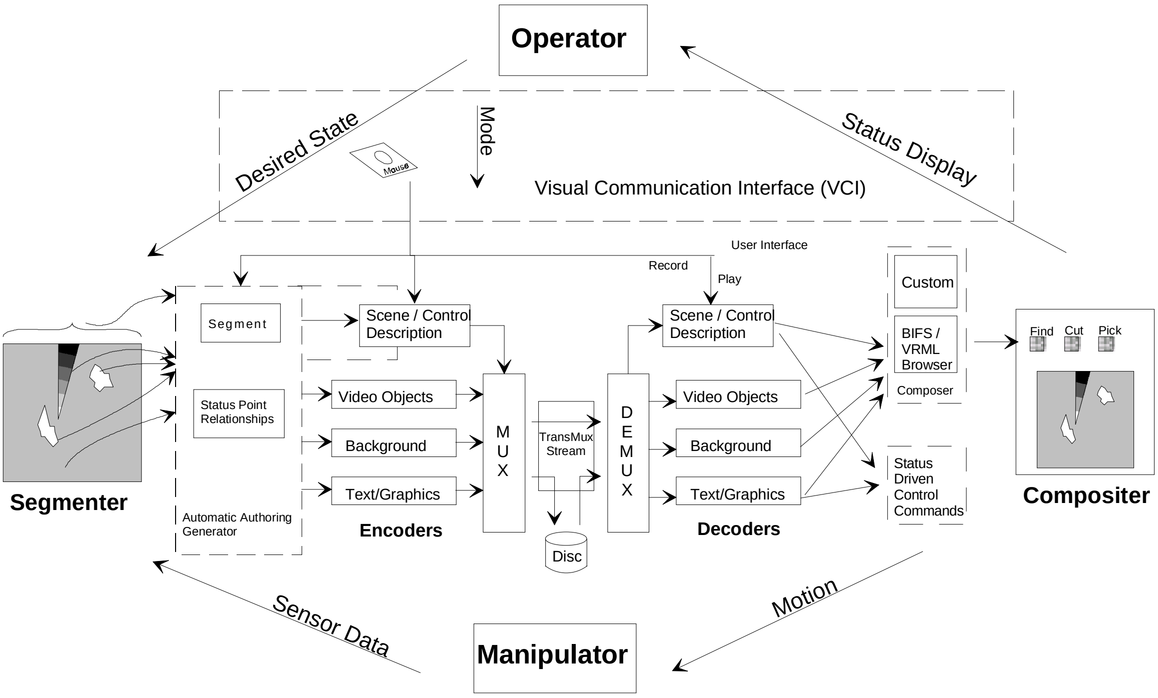


Figure 9. System Architecture

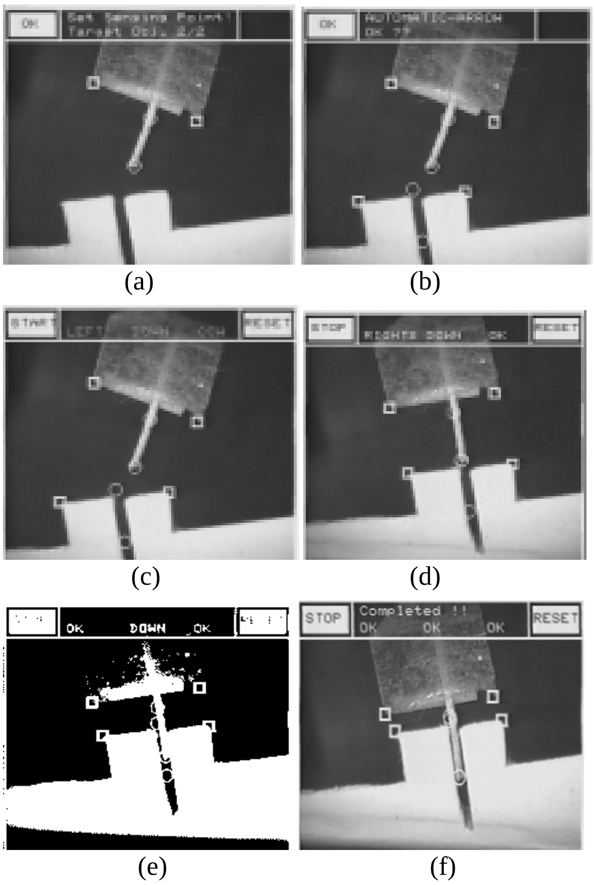
Although many schemes have been introduced to automatically segment an arbitrary video frame into meaningful objects, most have encountered only limited success or required highly constrained video sequences. The behavior sampling system exploits the teleoperated control nature of the video scene to aid in the segmentation of the video image into individual control objects. When an operator initiates a teleoperation sequence, he must specify the sensing points and mode/function of the system in order to describe the final state of his manipulation desires to the system. For the recording subsystem, segmentation of the individual video objects is also necessary. Although edge detection and morphological techniques alone can be used to provide proposed scene segmentation based on inter pixel contrast, it is desirable to relegate as many of the proposed objects to the background object plane to dramatically reduce the number of relevant objects to be tracked and encoded. The specification of the sensing points by the operator provides an efficient and non-intrusive means of separating relevant video objects by only selecting the object edges in proximity to the sensing points.

Spatial-temporal hierarchical object description of the scene follows the MPEG-4 scene description mechanism: Binary Format for Scenes (BIFS)[10]. Relevant groups of pixels in the image are grouped to form the target and environment object nodes. Compound sets of object nodes form tasks. Individual objects typically will have one or more sensing points associated with it. In this aspect, the sensing points are part of the spatial-temporal aspect of the scene even though the sensing points themselves are not part of the image representation. Although MPEG-4 provides anchor points and bounding boxes to reference a given blob as a video object, those anchor points typically have no relevance to manipulation surfaces on the object. In general, more than one pixel is necessary to describe a manipulatable characteristic of a target or environment object. For example, two sensing points may be used to describe an edge for placement.. BIFS is also encoded into a binary stream and multiplexed with the visual data stream.

EXPERIMENTS

Manual vs. Status Driven Task Execution

Two sets of experiments were conducted to contrast the performance of status driven operation and manual operation. For each task, two phases were required. In the teaching phase, the operator must configure the system for the desired movement function. In the status driven case, the operator must also specify the sensing points using the touch panel. In the second phase, the actual movement is performed. In the case of status driven control, the system controls and performs the movement. (see Figure 10) In manual control, the human manipulates the object by touching and dragging his finger in the direction of the desired motion.



(a) (b)

(c) (d)

(e) (f)

Figure 10. Insertion Sequence. Teaching phase (a) and (b). Initiation of automatic motion (c). Head to tail (d). Insertion (e). task completion (f).

The results of the first experiment with ten subjects were averaged and shown in Table 4. The results clearly show that the execution phase is performed more quickly when under status driven control. Unfortunately, the implementation of the current status driven teaching phase interface requires considerable time to specify all of the necessary sensing points.

It is also important to note in the last line of Table 4 that the status driven mode ensured the completion of the task. One of the operators during the manual insertion task was unable to complete the task before ramming the "peg" into the work environment.

Table 4. Comparison of Status Driven vs. Manual Operation for Placement and Insertion

	Placement		Insertion	
	SD	Manual	SD	Manual
Teaching Time (s)	43.1	6.0	40.1	5.2
Execution Time (s)	21.4	32.0	27.3	38.7
Total Time (s)	64.5	38.0	67.4	43.9
Completion (%)	100	100	100	90

Experiments in Cell Handling

Experiments with the behavior sampling system connected to the cell handling system (CHS) [11] were conducted. The CHS is capable of manipulating individual biological cells under optical microscopes. The scrapping tool effective width is approximately two micrometers and the translation accuracy is 0.5 micrometers.

The hardware used for the experiments includes: an Olympus BX60 microscope, 420-480 nm ultraviolet and white light sources, Sony XC-711 CCD camera, Matrox Genesis PCI image capture and processing board with TI 320C80 multi-DSP, Sigma mini-40XY pulse stepping motor stage, SMC-3(PC) pulse motor controller board, i686 MMX 266 MHz PC. (see Figure 11) The Windows NT 4.0 hosted software allows mouse based two d.o.f. control of the scraper by simply clicking on the captured image of the magnified work area.

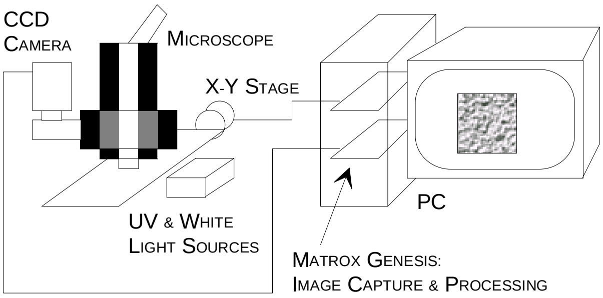


Figure 11. Block Diagram of Experiment system

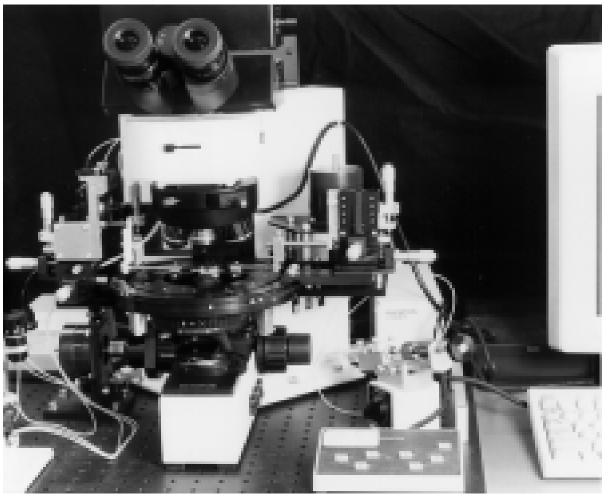


Figure 12. Microscope and Cell Handling System

The CHS is used in the preparation of single Mato fluorescent granular perithelial (FGP) cells for analysis. Mato FGP cells are found in the brain and studied for their relationship to aging and high-fat diets. Mato FGP cells, sometimes referred to as perivascular cells, are approximately 10 microns in diameter and exhibit an auto-fluorescent glow in the range 520 - 570 nm (green light) when exposed to ultraviolet light. This auto-fluorescent property is exploited in the image processing to help establish the approximate boundaries of the cell.

Individual cell analysis and manipulation is becoming increasing important for biological investigation. Heretofore methods of processing typically involved processing enmass without regard to the potentially disrupting effects of the tissue surrounding the cells. As individual cell manipulation is a developing field, there is very little human experience in such areas as how to manipulate the cell, tolerances of manipulations, appropriate tools, appropriate processes, etc. To help rapidly accumulate and exploit the new experiences and techniques currently being developed, the behavior sampling system is especially effective and being actively used. Biologists can maintain extremely accurate records of manipulation trials by visually reviewing the specific steps that a particular cell underwent. Status on demand functions can then be used to help recreate processes that were deemed effective.

One of the first examples of the Mato FGP cell processing is isolating the cell from the surrounding tissue. Although special lighting conditions combined with the auto-fluorescent property of the Mato FGP cell do provide good general cell boundary discrimination information, studies of various segmentation techniques are performed manually in order to observe the variances of the processing results. Thus behavior sampling is used to record the scraping path control information along with the visual information of the work environment.

Figure 13 shows the results of an operator manually scrapping around the cell. During the cell scraping, images of the process were captured along with the corresponding control movements. The behavior sampling system combined with the cell handling system has accumulated numerous cell manipulation trials.

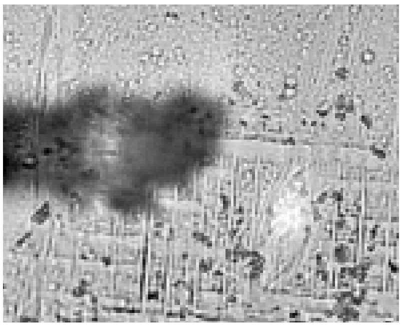


Figure 13. Results of manual scraping

Besides the long term analysis possible with behavior sampling, behavior sampling can be combined with status on demand to provide immediate assistance to the operator. Separating the cell from the surrounding tissue requires a band of sufficient width from the surrounding tissue. This band is wider than the scrapping area of the tool, thus several passes of the tool are required. A specialized form of the status on demand redo function has been developed to automatically scrape increasingly wider paths based upon the initial manual path. (See Figure 14 (right).)

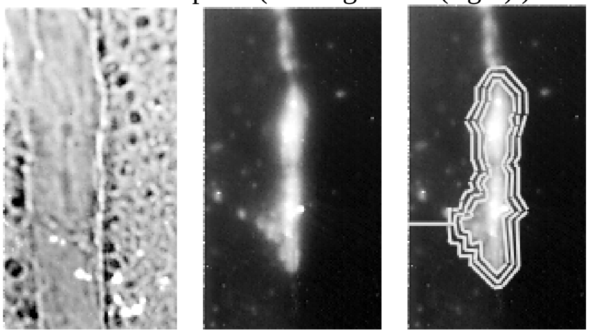


Figure 14. Cell illuminated with white light (left); Cell illuminated with UV light (center); Overlay view of scraping paths (right)

CONCLUSIONS

We have introduced the bilateral behavior media paradigm as an effective way of visually interacting for teleoperation. The status driven control method has been introduced and realized through the status driven micro handling system (SD-MHS). Behavior sampling allows the system to sample, structure, and store motion control sequences and their associated imagery. This behavior sampled data can be accessed to repeat or redo a recorded sequence.

Experiments have shown the effectiveness of the approach in both automatic and shared control modes in microworld manipulation tasks. Obviously, reducing the time of the status driven teaching phase would further improve the current implementation. Experiments using a cell handling system accumulated motion sequences of manipulating individual biological cells under optical microscopes.

Future work involves developing additional status on demand functions for general and specialized operator assistance. Finally, the status driven, behavior sampling, and status on demand components will be combined into a single system for efficient operator assistance.

ACKNOWLEDGMENTS

We wish to thank Takashi Miyoshi and Taichi Mita for use of the cell handling system and control software and the MPEG-4 IM1 team for their encoder and decoder software.

REFERENCES

[1] G. D. Hager, “A Modular System for Robust Positioning Using Feedback from Stereo Vision,” *IEEE Trans. On Robotics and Automation*, Vol. 13, No. 4. pp. 582-595, August 1997.

[2] N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade, “Visual Tracking of a Moving Target by a Camera Mounted on a Robot: a Combination of Vision and Control,” *IEEE Trans. On Robotics and Automation*, Vol. 9, No. 1, pp. 14-35, February 1993.

[3] T. Shibata, Y. Matsumoto, and T. Kuwahara, “Hyper Scooter: a Mobile Robot Sharing Visual Information with a Human,” *Proceedings of R&A 95*, Vol. 1, pp 1074-1079, 1995.

[4] T. Sekimoto, T. Tsubouchi, S. Yuta. “A Simple Driving Device for a Vehicle - Implementation and Evaluation,” *Proceedings of IROS 97*, Vol. 1, pp. 147-154, 1997.

[5] M. Mato et al, “Involvement of Specific Macrophage-lineage Cells Surrounding Arterioles in Barrier and Scavenger Function in Brain Cortex,” *Proc. Natl. Acad. Sci. USA*, Vol. 93, pp. 3269-3274, April 1996.

[6] R. Gonzalez, “Hypermedia Data Modeling, Coding, and Semiotics,” *Proc. of the IEEE*, Vol. 85, No. 7, pp. 1111-1140, July 1997.

[7] A. M. Murching et al, “Indexing Object Content Information (OCI) for MPEG-4 / MPEG-7,” ISO/IEC JTC1/SC29/WG11 M2878, Fribourg, Switzerland, October 1997.

[8] T. Sato, K. Koyano, M. Nakao, and Y. Hatamura. Novel manipulator for micro object handlingas interface bewteen micro and human worlds. *Proc.s IEEE/RSJ Inl Conf. on Intellignet Rob.s and Sys*, Vol 3, pp 1674-1681, July 1993.

[9] R. Koenen, “Overview of the MPEG-4 Standard,” ISO/IEC JTC1/SC29/WG11 N1730, Stockholm, July 1997.

[10] "Text for CD 14496-1 Systems," ISO/IEC JTC1/SC29/WG11 W1901, Fribourg, Switzerland, October 1997.

[11] T. Sato, T. Miyoshi, and H. Miyazaki, “Development of Cell Handling Robot System,” *Proc. Of RSJ 14th Meeting*, Vol. 3, pp. 1135-1136, November 1996. (In Japanese)