Human-Robot Interaction for Intelligent Assisted Viewing During Teleoperation

G. T. McKee, B. G. Brooks†
Department of Computer Science, The University of Reading, UK
gerard.mckee@reading.ac.uk
P. S. Schenker
Jet Propulsion Laboratory, USA
paul.s.schenker@jpl.nasa.gov

Abstract

The Visual Acts theory aims to provide intelligent assistance for camera viewpoint selection during teleoperation. It combines top-down partitioning of a task and bottom-up monitoring of the operator to select task-relevant camera viewpoints. Previous experimental studies have shown that Visual Acts provides camera views of sufficient quality to allow an operator to complete a task. In cases where the camera system is complex and difficult to master, it selects better viewpoints than the operator. In this paper we present an alternative architecture incorporating a viewpoint selection algorithm that places emphasis on what the operator should do next, rather than on what he is currently doing. Experimental results are presented showing that this simpler algorithm performs as well as the more pedantic Visual Acts algorithm, and raises greater awareness of the operator to 3D information. The results contribute to a better understanding of human-robot interaction in telerobotic scenarios.

1. Introduction

In *teleoperation* a human operator controls a robot located at a remote site [1]. A barrier, of space and time, exists between the operator and the robot that must be bridged for successful control of the latter by the former [2]. This paper focuses on one aspect of this divide, namely the visual inspection of the remote environment so that the operator can gather task-relevant information in order to guide the control of the robot. We are not concerned here with issues of time-delay, though an important aspect of remote operations, but solely with the control of cameras located in the remote environment to provide appropriate camera views for the operator.

In the traditional teleoperation setting the operator either has to manually control the cameras, in addition to controlling the robot, or a second operator controls the cameras [2]. The latter requires coordination between the two operators, typically using a prepared task script. Researchers have more recently approached the problem of control thorough various means based on immersing the operator into the remote environment (See [3] for a review). In our work we have emphasised the incorporation of more robotic intelligence. Our aim is to elucidate the character of the human-robot interaction required for intelligent assisted viewing of the remote environment, specifically providing mechanisms and models for automated camera placement.

Automating camera placement is successful to the extent that it reduces the complexity of the operator interface and provides views that allows the operator to extract task-relevant information as and when needed [2, 4, 5, 6]. The former essentially means reducing the need for the operator to manipulate the cameras and the latter essentially means having some technique that can determine where the cameras should be placed, or moved to, in order to improve the sensory data returned to the operator [7, 8]. In this paper we present further analysis of a theory, called Visual Acts, for automated camera placement [5, 9, 10]. The analysis complements and extends the results reported in [3, 11].

Visual Acts is a technique comprises two main components, namely a method for modelling and reasoning about the task being performed, and a method for selecting camera viewpoints that deliver optimal views in the context of a particular operation. Visual Acts considers a task at four levels of abstraction, namely task, subtask, visual goal and viewpoint evaluation levels [3].

†Currently with Brooks-Pri Automation, Reading, UK.

For example, an assembly operation, the task, typically comprises two subtasks, namely picking up an object and inserting it into an existing subassembly. The second of these operations is a kind of *peg–in–hole* operation [12]. Visual Acts would identify three visual goals for this second operation, namely:

1. The axes of the peg and hole must be parallel.
2. The axes of the peg and hole must be coincident.
3. The visible end-plane of the peg must be in the plane of the surface containing the hole.

Figure 1 shows the peg–in–hole task. Visual Acts uses the visual goals to direct viewpoint selection, via viewpoint evaluation functions, such that the views contain task-relevant information. Indeed, a significant portion of the Visual Acts model is the selection of camera viewpoints to deliver task-related data to the operator. This includes monitoring the operator during a task in order to select which assembly subtask is active, which visual goal is active, and evaluating viewpoints with respect to the visual goal.

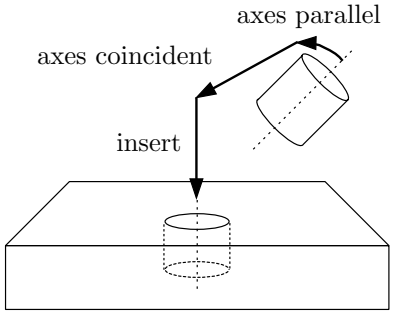

Figure 1: Visual Goals for Peg–in–Hole Operation.

In a previous paper we reported experimental results confirming that Visual Acts provides camera views of sufficient quality to allow an operator to complete a task, and indeed improves on manual control as measured by time for task completion [3]. In a further paper we presented results aimed at elucidating some of the reasons why Visual Acts works [11]. The main conclusion was that where the camera system is complex and difficult to master, Visual Acts will select better viewpoints than the operator. In addition, even removing the time taken to control the camera system, Visual Acts still significantly reduces the length of time manipulating the objects in the remote workcell. These results are attributed to the relative improvement in viewpoints that Visual Acts selects versus the operators own selection. However, since the operator is surely able to conceive of good viewpoints, it seems that the operators gives up trying to maneuvre the camera into good positions. In other words, it appears that the operators are far more tolerant of viewpoints than the Visual Acts model gives them credit for.

In this paper, therefore, we report on a further set of experimental results based on a simpler algorithm for viewpoint selection. This simpler algorithm is not meant as an improvement or a refinement of the Visual Acts algorithm. It is based in large part on the simple algorithms of AUTOVIEW [13] and ADEPT [14], but incorporates the hysteresis element of the Visual Acts model [3]. Its aim is to elucidate further the relative roles and contributions of the operator and the robot in human-robot interaction for telerobotic tasks [2]. This algorithm also places emphaisis on what the operator needs to do versus what the operator is currently doing. Hence, it can be much less pedantic than the Visual Acts algorithm.

The remainder of the paper is organised as follows. Section 2 provides a review of the Visual Acts model as implemented in previous experiments. Section 3 discusses a number of relevant observations made during the initial set of experiments that evaluated the Visual Acts implementation. Section 4 outlines the alternative viewpoint selection algorithm. Section 5 presents the experiments conducted to assess the new algorithm, the results obtained and their anlysis. Finally, section 6 provides a summary and conclusions.

2. The Visual Acts Implementation

The Visual Acts implementation comprises a deliberative and a reactive component. The deliberative component decomposes the high-level task into a set of atomic operations, then decomposes these into visual goals, and represents the latter as viewpoint evaluation functions in the viewpoint selection algorithm. The decomposition does not rule out any possible variation: it neither dictates the sequence of operations nor the way in which operations are performed. The functions of the deliberative component were executed off-line. The reactive component concentrates on monitoring the operator to discover which task is being performed and which visual goal describes the operator's current actions, and evaluating proposed viewpoints.

Visual Acts starts out with three quantities that are given before the task can begin:

1. A geometric model of the remote workcell.

2. A kinematic model of the camera placement system, the remote teleoperated manipulator(s), and an optical model of the cameras.

3. A geometric description of the task.

The geometric model is manually calibrated to the real environment.

The Visual Acts model evaluates visual goals using a technique, visibility modelling, that assesses the degree of task completion that can be observed in the image. Visibility modelling aims to find viewpoints that ensure *sufficent* information for the operator to localize the objects, but also to optimize the viewpoint for evaluating the information pertaining to the visual goals. A more detailed description of the visibility modelling is presented in [3]. The visibility modelling is incorporated within the Visual Acts search algorithm for viewpoint selection.

The viewpoint search algorithm uses pre-selected seed points that are known to be within the constraints of the camera manipulator kinematics, visibility of the *workcell*, field–of–view and depth–of–field of the camera. These viewpoints are deliberately chosen to cover the entire region as evenly as possible. A separate hill-climbing search is started for each pre-selected seed point and is restricted to a limited volume designed to reduce unnecessary overlap with neighbouring searches. Each search executes each test roughly in the order of computational cost and selectivity. The tests for camera manipulator kinematics, visibility, field–of–view and depth–of–field simply return a boolean value indicating whether the viewpoint meets the minimum requirements of the constraint. If rejected by any test, the search is immediately discontinued in that area.

The architecture developed for the Visual Acts implementation was a multi-agent architecture based on the agent architectures of [15] and [16]. In the Visual Acts architecture every visual goal, viewpoint evaluator, and camera resource is modelled as an agent. The **visual goal agents** monitor the operator to see how closely the operator's actions match those suggested by their visual goal. Each **viewpoint evaluator agent** searches for a viewpoint that provides the most information relevant to their visual goal. The **camera resource agents** collect the proposals from all the viewpoint evaluator agents. They in turn post the full set of proposals, along with the current viewpoint, back to the viewpoint evaluator agents. The latter now evaluate each proposed viewpoint, and the current viewpoint, with respect to their own requirements, and return the results to the camera resource agent. The camera resource agent sums the scores for each proposed viewpoint. It selects the one with the highest aggregate value and compares it to the value of the current viewpoint. The camera is then instructruted to move to the winning viewpoint.

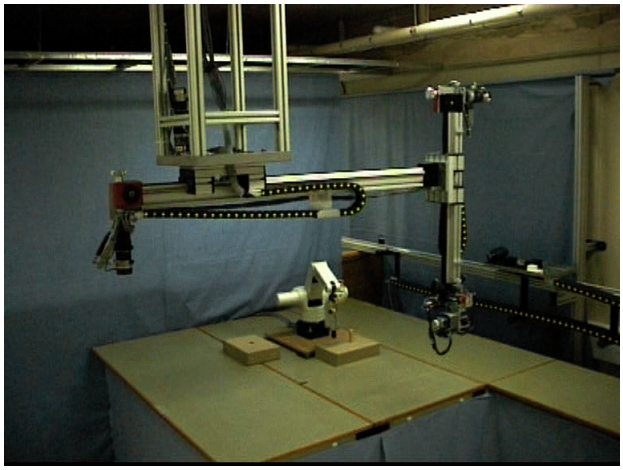

Figure 2: The laboratory setting for the Visual Acts experiments.

The Visual Acts algorithm also incorporates an hysteresis element to reflect the fact that operators tend to prefer stability of the camera over the value of the viewpoint. The hysteresis elemented weighted the current viewpoint against the proposed viewpoint. The longer the current viewpoint is held, the more amenabile it becomes to change, and the more persistent the proposed viewpoint is the more likely the camera viewpoint will change. Finally, the method of determining the operator's current intentions is to predict for each visual goal how the operator would behave were he to perform the task exactly as promoted by the goal, and then to measure how well this prediction matches his current activity. The latter is evaluated based on the motion of the object with respect to the target, which equates to the motion of the manipulator end effector.

For the experiments reported here, and those reported in [3, 11], a single camera mounted on a gantry robotic crane was used to provide viewing of the task environment. Figure 2 shows a view of the laboratory setting for the experiments. The experimental task consisted of grasping a wooden peg and placing it in a hole. The experimental environment included a mobile camera and a robotic manipulator. The former comprised a black and white camera on a pan-tilt head that was mounted on the crane. The camera had automatic iris control and a fixed–focus lens, leaving the operator in control of the crane and the pan/tilt motion

of the camera. The robot manipulator was a 5 degrees of freedom CRS robot arm.

3. Observations on Visual Acts

Although the results of the experiments that evaluated Visual Acts were encouraging, some observations suggested weaknesses in the algorithm. These can be traced to a serveral aspects of the human operators' and Visual Acts' performance:

- The influence of environmental constraints on viewpoint selection.
- The effect of automatic camera placement on the operator's sense of teleproprioception.
- The effect of moving the camera, and camera motion, on operator control.

In addition to the task–based requirements of the viewpoint, the placement of cameras in a real robot environment is constrained by a number of other factors. Visual Acts is forced to take into account the kinematics of the camera manipulator and the optical properties of the camera. Observations from the experiment suggest that the camera system, in spite of its flexibility, constrained viewpoint selection considerably. Two factors were particularly influential:

1. The physical construction of the camera manipulator precludes large volumes of space in which Visual Acts might have considered viewpoints. For example, the crane cannot raise the camera high enough to clear the robot arm and adopt viewpoints looking down from above.
2. The fixed focus and small viewing angle of the camera forced Visual Acts to keep the camera a long way from the workcell. Since many of Visual Acts' evaluation functions depend on the relative distances and angles between features in the image, and the prominence of those features, Visual Acts favours viewpoints that are much closer to the workcell. The effect is that the optical constraints of the camera prevent the search function from considering viewpoints valued highly by Visual Acts.

Further, the operator is generally much more tolerant of poor viewpoints than Visual Acts. This was evident in the poor viewpoints chosen by the operators and also in their tolerance of out–of–focus views. The inclusion of hysteresis in the viewpoint selection algorithm explicitly acknowledges this fact: the operator prefers camera stability to view quality. The viewpoint is deliberately held for a period of time regardless of other suggested viewpoints that may be better, in order not to overwhelm the operator with rapid viewpoint changes. Operators must be able to (and in the case of manual camera control, they do) cope with viewpoints with a range of quality.

Finally, Visual Acts tries to detect which visual goal the operator is currently concentrating on, in order to direct the selection of viewpoints. It does this by monitoring the operator and matching the operator's control of the manipulator to the motions expected for each visual goal. The degree of match for each visual goal indicates to Visual Acts how much that visual goal is "active". This reactive component, however, has two flaws:

- The operator is effectively forced (due to the complexity of the manipulator controls) to manipulate one degree of freedom at a time. This can lead to Visual Acts mis-interpreting the operator. The solution adopted was to measure motion of the arm over longer periods of time, smoothing away the path that the manipulator has taken during that time. This is a poor solution since it reduces the sensitivity of Visual Acts, and introduces a new variable in the system, the setting of which can drastically change the behaviour of Visual Acts.
- The emphasis on Visual Acts from its earliest inception was to monitor the operator and respond to his demands for information. The reactive component of Visual Acts was designed to do this and alter the viewpoint to meet these requirements on demand. The experiments, however, showed quite clearly that the operator is very much influenced by the *current* viewpoint. He tends to select his task based on the information currently available in the image. This was shown up particularly well by the surprise that operators expressed when Visual Acts changed viewpoints, and the operator saw how poorly aligned the gripper was.

The experiments, therefore, have shown that Visual Acts does select sensible and valuable viewpoints, and does improve operator performance. The observations, however, suggest that Visual Acts may be too good. Not that Visual Acts is wrong, but that the coarse constraints of kinematics and optics, and the aim of maintaining camera stability at the expense of viewpoint quality, suggest that a simpler algorithm would suffice. In summary, these several observations have suggested:

1. Visual Acts is too pedantic in its choice of viewpoints. Especially since the physical constraints of the environment prevent this very detailed algorithm from being exercised fully.

2. The operator is more tolerant than Visual Acts expects.

3. It might be sensible for Visual Acts to pay as much attention to what the operator *needs* to do, as it currently does on what it thinks the operator *is* doing.

4. Viewpoint Selection

Visual Acts deliberately set out to analyse the teleoperation task from a high level model–based approach. It attempts to achieve good viewpoints based on its ability to reason about the task, reason about the operator's current intentions, and its ability to control and manœurve the camera into position. The observations discussed above suggest that a simpler algorithm for viewpoint selection would suffice. In this section we describe an alternative viewpoint selection algorithm which we refer to as the *reactive camera placement algorithm*.

This algorithm directly relates the current position of the objects in the environment to good viewpoints. It has no concept of how the operator is modelling the task in his mind, the techniques that the operator might use to overcome the lack of proprioceptive feedback, or the complexity of the manipulator controls. Its only concession is the use of a hysteresis function to prevent rapid or erratic viewpoint changes.

The reactive algorithm is not meant to be an improvement over Visual Acts. It does not try to correct the weaknesses exposed by the previous set of experiments, nor to tune Visual Acts to perform any better. It is a simple replacement based on the simple algorithms of AUTOVIEW [13] and ADEPT [14]. The original intention was in fact to run the experiments again to test Visual Acts against one of these systems. However, both systems depend to a large degree on very specific scenarios and could not be applied unmodified. Therefore, a new viewpoint selection algorithm was developed.

One fundamental difference between the two algorithms is that Visual Acts tries to provide views that it reasons the operator wants, whereas the simpler algorithm concentrates solely on views that it believes the operator needs. They both essentially aim to provide the same information, but the difference of approach manifests itself in the way the algorithms react to the operator's actions. One of the fundamental strengths of Visual Acts has been incorporated in the reactive algorithm, namely hysteresis. Hysteresis helps the operators to maintain (to some degree) their sense of teleproprioception.

The viewpoint selection algorithm of AUTOVIEW [13] uses the object being moved (manipulator), the target object, and the two closest obstacles, to compute a viewpoint (figure 3). The viewpoint provides a view that has the distances from the manipulator to the other objects perpendicular to the line of sight. There are two difficulties with applying this algorithm exactly as described:

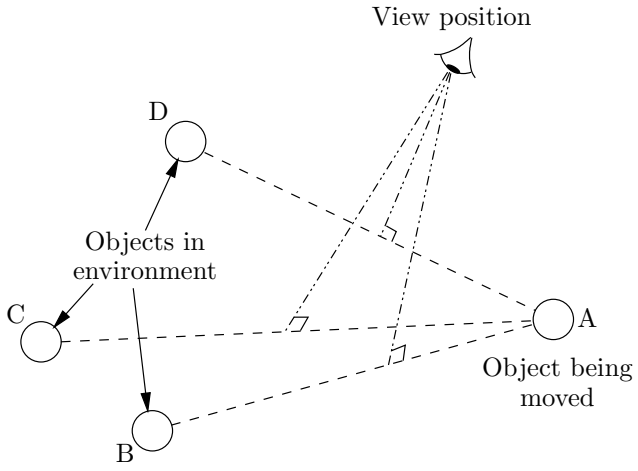

Figure 3: View of distance of objects from moving object. Reproduced from [13] page 4.

1. The algorithm cannot be combined with the other physical constraints of the environment without being re-written as a viewpoint evaluation function.

2. The algorithm explicitly includes the two closest obstacles. As the manipulator moves past obstacles the set of closest obstacles changes. The effect is that the viewpoint appears to the operator to move erratically.

The new viewpoint selection algorithm continues with the basic idea, to maximize the position error in the image, but drops the use of the obstacles to constrain the viewpoint. The new selection algorithm only considers

the virtual line between the object and the target, and the plane that bisects that line. Viewpoints are valued highly that minimize the angle between the view orientation and the plane (θ in figure 4).

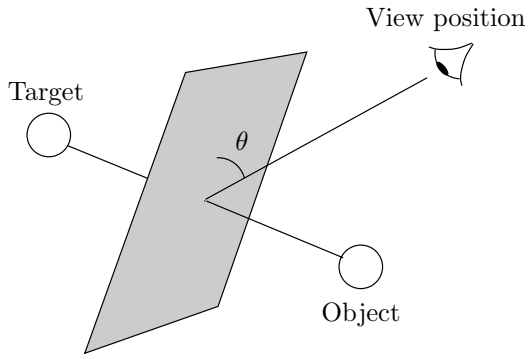

Figure 4: New viewpoint evaluation function.

An alternative way of representing this concept is to generate a computer simulated image and maximize the distance between the object and the target in that image. The distance between the objects in the image is proportional to $\cos(\theta)$ (where θ is the angle in figure 4) and to $depth^{-1}$ (where *depth* is the distance of the camera to the objects). Maximizing this function orients the camera orthogonally to the *task* (as is done explicitly by the AUTOVIEW system), and moves the camera towards the objects, increasing their resolution and prominence in the image.

The work on ADEPT made two further recommendations [14]:

- The camera roll should always be set such that a consistent reference frame is maintained with respect to the manipulator. This constraint is already enforced by the camera manipulator — the camera cannot be rolled.
- The target should be in the centre of the image. This recommendation was made based on feedback from the subjects who took part in their experiments.

Therefore, the new algorithm also includes a term that tries to minimize the distance from the target object to the centre of the image.

The same very primitive search function as used by the Visual Acts experiment was implemented. That is, given a set of seed viewpoints known to be "reachable" and within the optical constraints of the camera, a simple hill climbing search is performed. The evaluation function was as follows:

1. The proposed viewpoint was tested for "reachability" (camera manipulator kinematics), and optical constraints (depth–of–field only). If either failed, the viewpoint was immediately rejected.
2. A computer simulated image of the workcell from the proposed viewpoint was generated.
3. The visibility constraint was tested. The object and target had to be visible in the image (self occlusion excepted). If either object was occluded by an obstacle then the viewpoint was immediately rejected.
4. The distance between the object and target in the image, d, and the distance of the target from the centre of the image, c, were measured. The value of the viewpoint was given as:

$$V(v_p) = \frac{a\frac{d}{s} + b\left(1 - \left(\frac{c}{s}\right)^2\right)}{a+b} \qquad (1)$$

where s is the size of the image (measured diagonally), and a and b were constants used to tune the function (both set to 1).

As in Visual Acts, the algorithm also included hysteresis to prevent the viewpoint from being moved too quickly (at least not in the first t seconds), and then only if the new viewpoint improves over the current viewpoint.

$$v = \begin{cases} v_{current} & : \text{ if } \frac{tV(v_{current})}{time} > V(v_{new}) \\ v_{new} & : \text{ otherwise} \end{cases} \qquad (2)$$

where $V(v_p)$ is the value of a viewpoint, *time* is the length of time in seconds that the current viewpoint has been held, and t was initially set to 3 seconds.

5. Experimental Results and Analysis

The experimental evaluation of the new algorithm was conducted using exactly the same set–up as for Visual Acts, allowing direct comparisons to be made between the two. The intention again was to extract timing information from the subjects about the speed of operation and the length of time spent operating the manipulator, and to measure performance based on those timings.

Only the first part of the Visual Acts experiment was repeated: the task was to move the gripper to pick up

a peg standing on a box. The gripper was constrained to open only to an extent of 1.8cm, and mounted on the end of the CRS robot arm. The peg was a wooden block with dimensions 7.5cm, 2.5cm, and 1.5cm, initially set half a metre from the gripper. The operator was given a single view from a single camera mounted in the environment (see figure 2).

Emphasis was placed on avoiding collisions and accuracy of manipulation. The task was performed twice, once using Visual Acts, and then again using the reactive viewpoint selection algorithm. Subjects were given time to become acquainted with the controls and to "play" with the system until comfortable (typically ten to fifteen minutes). To reduce the effects of practice and fatigue on the results, subjects were randomly assigned to two groups. One group performed the task using Visual Acts first, and the other performed the task using the new algorithm first.

In this study six subjects were chosen at random. All male students of the Departments of Cybernetics or Computer Science, aged between 20 and 30. Given the limited amount of training time, it was decided that it would not be possible to train the subjects with both automated camera placement algorithms, and that using only one or other might unfairly influence the results. Therefore, during training, the camera was set a fixed distance from the manipulator giving a wide angle view of the workcell. The subjects were not given control of the camera manipulator. They were only able to control and practice moving the robot arm.

Once every half second throughout the test, the positions and velocities of all the robot joints in the laboratory, and the velocity instructions from the operator's console, were recorded in a log file. Timings were extracted from the log file, and are shown in table 1. This

Subject	Visual Acts	Reactive Algorithm
1	261.5	168.0
2	149.0	189.0
3	224.5	139.5
4	198.0	225.5
5	134.5	125.5
6	170.5	211.0

Table 1: Times to completion.

information is also shown as a bar chart in figure 5 and when fitted to the normal distribution gave a mean of 176 and a standard deviation of 39. The results for Visual Acts control are comparable with those for the Visual Acts control in the previous experiments [3].

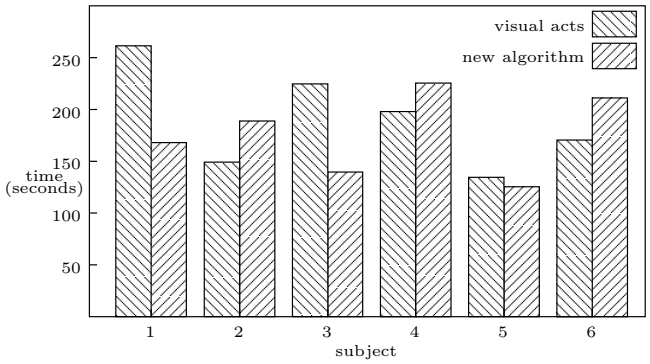

Figure 5: Timing measurements

The timings were compared using a two tailed t-test. The t-test compares the differences between the two data sets and indicates whether the mean of the differences can be explained by the null hypothesis. This test was performed and a value of 0.5266 calculated for t_{obs}. The null hypothesis cannot be rejected. The difference in times observed between the two algorithms could have occurred by chance.

Mean of the differences	Standard Deviation of the differences	t_{obs}
13.25	61.6301	0.5266

Table 2: Comparison of Visual Acts against the reactive algorithm.

The results, for this experiment, show that Visual Acts does not out–perform the simpler reactive algorithm. This confirms predictions that the most important factor in improving operator performance was relieving the operator of the task of controlling the camera manipulator. The benefits of providing *good* viewpoints was secondary. Our expectation was that any algorithm that could take care of camera placement regardless of its ability to select high quality viewpoints would immediately raise the importance of other factors in the experimental results. The importance of camera viewpoints is likely to be a combination of the complexity of the task and the experience of the operator:

- If the task is too difficult then novice operators have difficulty managing the manipulator.
- If the task is too easy then camera placement loses its importance and allows other simpler algorithms to compete favourably.

In this case the reason that *good* viewpoints are not emphasized in the results is likely to be due to the simplicity of the task rather than indifference of the operator. The task was deliberately chosen to be easy because the subjects were untrained.

This appears to be bourne out by the following further observation. The method used (in both experiments) to reduce the effects of practice on the results was to choose operators at random. "Odd numbered" subjects performed the task using Visual Acts first, and "even numbered" subjects used the reactive algorithm first. Looking again at the results plotted in figure 5 it is clear that every subject performed the task faster in their second attempt regardless of which algorithm they used. This is shown graphically in figure 6.

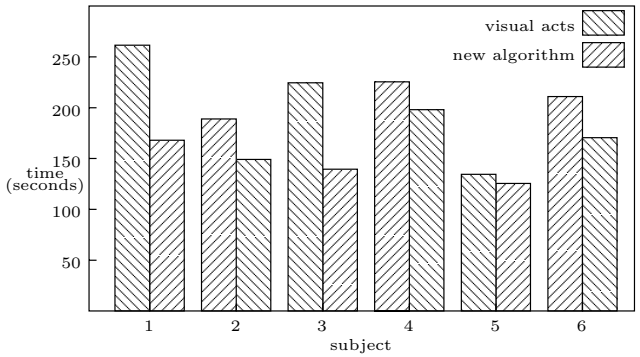

Figure 6: Timing measurements given in the order they were performed.

The viewpoints selected by each algorithm for one subject (the fifth subject) are shown in figure 7. The most obvious pattern that can be seen in this diagram is the even distance of the viewpoints from the workcell. This supports one of the points made earlier: that the physical constraints are the most influential.

The most interesting observation from the experiment is the types of viewpoint changes made. The reactive algorithm made more viewpoint changes and larger viewpoint changes than the Visual Acts algorithm. This reflects their different responses to operator actions. Visual Acts chooses to maintain viewpoints until the operator clearly shows that he has changed task, whereas the reactive algorithm changes viewpoints as soon as another showing more position error becomes available.

With the reactive algorithm, as the operator reduces position error in the image (in two dimensions), the error in the third dimension (represented by depth in

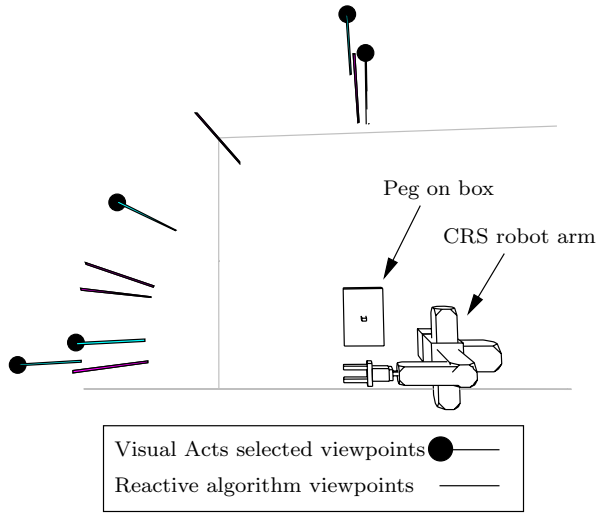

Figure 7: Viewpoints for one subject.

the image) becomes relatively large. The algorithm changes to show this error as soon as it becomes larger than the error in the image (plus waiting time introduced by the hysteresis). There are two effects as this process continues:

1. Since the operator can only effectively reduce whatever error is presented to him in the image, he is forced to move the manipulator more directly towards the target.

2. The frequent orthogonal viewpoint changes keeps the lack of depth in the image and its corresponding problems firmly in the operator's mind. This indirectly encourages better performance since the operator has to be more considerate about depth and how the object is moving in space, as opposed to just considering the movement of the object in the image.

These observations support the earlier contention that viewpoint selection is not the weak part of Visual Acts; it's the reactive component.

Finally, the use of untrained operators appears to have influenced the results, obscuring the differences between the two algorithms tested. It suggests that when this type of task is performed by untrained operators, it will likely benefit similarly using either algorithm. The benefit being the removal of the burden of camera control. The factors that prevent Visual Acts from performing better than the simpler algorithm are *1*) that the simplicity of the task (demanded by the

novice operators) does not call for well placed viewpoints, and *2*) the restricted reach of the camera manipulator prevents Visual Acts from offering its full range of viewpoint choices.

Future experiments would benefit from using trained operators as subjects. The most interesting conclusions, however, come from the qualitative results — that the reactive component of the new algorithm prompted the operator more forcefully to consider depth. Perhaps, therefore, Visual Acts should be as assertive.

6. Summary and Conclusions

In summary, Visual Acts is a model for automated camera control during teleoperation. A key aspect of its operation is its interaction with the human operator. Its role is to provide camera views in a timely manner to satisfy the needs of the operator for task-relevant information. Its chief benefit is the removal of the burden of camera control from the operator, who is left to concentrate on the manipulation component of the task. Previous results showed that Visual Acts does indeed provide viewpoints that the operator can use, as measured by time to completion of a peg-in-hole task. Observations made during the experiments to evaluate Visual Acts suggested that Visual Acts is too pedantic. It focuses too much on what the operator is currently doing and not enough on what the operator needs to be doing. In this paper we have presented an alternative algorithm which places emphasis on the latter. It is a simpler viewpoint selection algorithm than the Visual Acts algorithm, but incorporates elements of the Visual Acts algorithm, most notably a hysteresis component. The experiments reported in this paper show that this simpler algorithm performs as good as the Visual Acts algorithm. Even thought it incorporates a hysteresis element, it is much more assertive than Visual Acts. It forces the operator to have a much greater awareness of depth information. The results also suggest that futher experiments would benefit from the inclusion of trained subjects if the differences between the two algorithms are to be elucidated in more detail.

References

[1] Thomas B. Sheridan. *Telerobotics, Automation, and Human Supervisory Control*. MIT Press, 1992. ISBN 0-262-19316-7.

[2] Paul S. Schenker and Gerard T. McKee. Human–machine interaction in telerobotic systems and issues in the architecture of intelligent and cooperative control. In *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 385–390, Monterey, California, August 1995.

[3] B. G. Brooks, G. T. McKee, and P. S. Schenker. The visual acts model for automated camera placement during teleoperation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1019–1024, Seoul, Korea, 21–26 May 2001.

[4] Yujin Wakita, Shigeoki Hirai, and Toshiyuki Kino. Automatic camera–work control for intelligent monitoring of telerobotic tasks. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1130–1135, Raleigh, North Carolina, 7–10 July 1992.

[5] G. T. McKee and P. S. Schenker. Visual acts for goal–directed vision. In Paul S. Schenker, editor, *Sensor Fusion VII*, volume 2355 of *Proceedings of the SPIE*, pages 314–322, Boston, Massachusetts, 31 October–1 November 1994.

[6] Hari Das, Thomas B. Sheridan, and Jean-Jacques E. Slotine. Kinematic control and visual display of redundant teleoperators. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 1072–1077, Cambridge, Massachusetts, 1989.

[7] Konstantinos A. Tarabanis, Peter K. Allen, and Roger Y. Tsai. A survey of sensor planning in computer vision. *IEEE Transactions on Robotics and Automation*, 11(1):86–104, February 1995.

[8] Seth Hutchinson, Greg Hager, and Peter Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, May 1996.

[9] Gerard T. McKee and Paul S. Schenker. Visual acts for remote viewing during teleoperation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 53–58, Nagoya, Aichi, Japan, 21–27 May 1995.

[10] Gerard T. McKee and Paul S. Schenker. Human–robot cooperation for automated viewing during teleoperation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 124–129, Pittsburgh, Pennsylvania, 5–9 August 1995.

[11] B. G. Brooks, G. T. McKee, and P. S. Schenker. The visual acts model for automated camera placement during teleoperation: Further results and analysis. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1019–1024, Washington D.C., USA, 21–26 May 2001.

[12] J. L. Nevins and D. E. Whitney. Research on advanced assembly automation. In *IEEE Computer*, pages 24–38, December 1977.

[13] Hari Das. Automatic visual sensor placement for teleoperation, 1991. Unpublished. See [6] and [1].

[14] Gregory M. Pisanich, Michael P. Prevost, and Steven B. Hall. Evaluating the impact of camera placement on teleoperator efficiency. In *Proceedings of the First International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pages 629–636, 1988.

[15] Luís Correia and A. Steiger-Garção. A model of hierarchical behaviour control for an automated vehicle. In P. Gaussier and J. Nicoud, editors, *From Perception to Action*, pages 396–399, Lausanne, Switzerland, 7–9 September 1994. IEEE Computer Society Press. ISBN 0-8186-6482-7.

[16] J. K. Rosenblatt. DAMN: A distributed architecture for mobile navigation. In H. Hexmoor and D. Kortenkamp, editors, *Proceedings of the AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents*, Stanford, California, 27–29 March 1995.