A Cooperative and Visualized Environment for Virtual Experiment

Maiga Chang, Ivory Chung, Yu-Wei Jeng, Jia-Sheng Heh
Dept. of Info & Comp. Eng., Chung Yuan Christian Univ., Chung Li, Taiwan

Abstract

This paper proposes a virtual experiment environment for integrating those previous studies, such as cooperative learning, visual learning and situated learning, into the traditional practicum, which is used to verify the learning knowledge of class by manipulating virtually and well defined in the conventional curriculum design. For accomplishing the cooperative task in a real laboratory, the virtual environment should be constructed on the computer network.

keyword: V.E.E. (Virtual Experiment Environment), experiment model, knowledge object, activability, verificableness

1. Introduction

After conventional education has proceeded for thousand years, the pedagogical system is now greatly impacted by the new computer technology. Besides the computer brings the assisting system into instruction and brings intelligence into tutoring, the Internet also drastically changes our educational environment. Such that, a virtual environment for tutoring the practicum in the traditional school is also needed to design on the computer network for training students. And of course, the important issues of practicum, visualization and manipulation, shall not be lost in the virtual environment by the previous research of visual learning and situated learning.

Besides, experiments also stress the property of real-time, which means that the processes and results of experiments can be observed by learners immediately. Generally speaking, there are two phases in a virtual experiment, simulation and verification: the former phase is to simulate the process of experiment with the laws of nature, and the latter one is to verify the knowledge which derived by learners.

In Section 2, the learning theories in the computer science are tried to bring into the practicum of conventional education, and then the analysis model of experiment is constructed in Section 3. Section 4 designs an architecture of a virtual experiment environment. An experimental system is implemented with a real experiment of physics in Section 5. Section 6 is a conclusion.

2. Brings learning theories to practicum

As the computer technology develops rapidly for decades, CAI (*Computer Assisted Instruction*) is investigated by many psychologists and educators to support traditional education. The role that computer plays was usually to be designed as intelligent tutor, which means that artificial intelligence is involved and bring CAI into ITS (*Intelligent Tutoring System*). The goal of ITS is to simulate the computer as an intelligent tutor which can interpret user's behaviors and react appropriate tutoring strategies.

Computers not only replace the communication between teachers and students in traditional education, but also extend the learning curriculum to be remotely cooperative and competitive through communicative learning media. A curriculum which can be divided into several parts and achieved by independent works of individuals, called cooperative curriculum. Whereas, the competition in a learning system is to construct a way to let learners have strong attempt to compare thelearning performance with others'

Besides being an intelligent tutor and cooperation/competition media, computer can also be a visualized demonstration platform to process the visual learning. [3] *Visual learning* presents the knowledge structures or learning procedures through some software visualization methods such that the learners can catch the concept visually and then the tutor and learner can point out what the knowledge formation will be. The technology of software visualization involves algorithm visualization and program visualization. The former includes static graphics algorithm and animation algorithm, such as animation of sorting algorithm.[1] The latter is program visualization, which utilizes 2D/3D graphics to exhibit the data and code during program execution. There are many previous researches about the visualization of LISP programming, including a visual tracer [6] and a cooperative and competitive learning system "TurtleGraph". [3]

Considering the practicum of conventional education, John Dewey proposed another critical idea of "learning by doing", which then brings *situated learning* as the interactions of human cognition are studied and improved in different computer-based learning environment. The theory of situated learning has several essential points.(Brown, Collins & Duguid, 1989) First, the theory emphasizes that learning is determined not only by individual pre-stored knowledge, but also by the learner's understanding about each action in the situation of the practicum. Second, the theory stresses that knowledge should be practiced to make learning more meaningful. Such learning strategy utilizes the technology to build a "learning community" to enculturate the learners with distributed expertise and social discourse.

After the investigation of the above learning theories, it is possible to utilize these theories in practicum, especially for building such environment on computer network. First of all, such an environment should provide a situated learning environment for the learners to practice and then learn through the manipulation. Secondly, this environment visualize its knowledge and procedures to make the learners imagine to live in such environment virtually. Finally, the previous studies showed that the learning rate of students can be increased by the cooperative sanction and there will be multiple solution for the same question through panel discussion among participants. The environment designed for practicum should be a society in which the learners can acquire the knowledge through certain social behavior, cooperatively or competitively.

Building an environment for a practicum simulated on the computational interconnected machine and visualization interface is to construct a *virtual experiment environment* on computer network. A *virtual experiment environment* provides a visualized social learning environment with creating experiment situation on Internet. It is possible to list the advantages and disadvantages of such environment. The advantages are:

- **money saving:** for equipment preparing with user defined equipment.
- **time saving:** for experimental scene arrangement with visualized tools.
- **resource reusability:** This is because resource on the Internet can be reused, e.g., equipment and visualized tools.
- **time and space distributive:** Learners can learn knowledge through computer network.
- **realization of expansive, impossible experiment:** supported by computer and Internet, e.g. frictional problem, Kepler's Planet Law.
- **easy data recording:** The computer can run the experiment and provide experimental data with accuracy.
- **concentrated learner focus:** Since the experimental data can be obtained from computers, learners can focus on the knowledge of specific practicum.

Also, there must exist some deficiencies, which could be:

- **lack of the feeling of the real world.**
- **difficulties in represent the accuracy of experiment.**

Following the previous comparison, the V.E.E.has to be proven to possess enough advantage. Hence, It is worth designing the V.E.E. to assist the learners to learn from the experiment. The next section will analyze such environment according to the model of real experiment.

3. Analysis model of experiment

As the principle of school education indicated (Bobbitt, 1924), the goal of school education does not fall on the general abilities which would be learned in one's usual journey, but focuses on those kinds of knowledge that are complicated enough and can not be developed during daily life. Formally, we can express a piece of such analytic knowledge as K ö K, where K is the universe of related analytic knowledge pieces. Moreover, the knowledge piece that the learner actually obtains in this study could be denoted as $\overset{ì}{K}$ (ö K), an knowledge estimate of knowledge piece K .

Working process of experiment

The development of knowledge in school education always relies much on the "implementation," which is then an important character of Bobbitt Education Mechanism. (Bobbitt, 1918) This rouses a school to try her best to provide her students with chances of exercising, such as doing the experiments.

In the general discussion of courseware design, an *experiment* is designed with one kind of technology and useful knowledge to manipulate and fire away at the school education system. (Gagne, 1968) [2] Based on the above study of cooperative education and situated learning, an experiment can be decomposed into five working stages, as shown in Figure 1, with each stage explained in the following.

1. **Previous Test:** There are four major factors for making up an efficient course: sequencing, continuity, integration and articulation. (R. Tyler; Oliver, 1977; Ornstein & Hunkins, 1988) [2] The function of previous test in an experiment is to check the background knowledge of its participants with several related subjects for keeping the sequence and articulation of the practicum.
2. **Task Assignment:** During the process of an experiment, there would be lots of various tasks, such as equipment preparation, experimental scenes arrangement, data recording. The participants can acquire the knowledge not only through what is manipulated, but also through what would be observed; therefore, the tasks should be distributed before an experiment is processed.
3. **Experiment Conduction:** Two things should be taken into consideration in this stage: the realization of nature laws and the acquisition of the results. For example, the starting of a physics experiment would make the nature law work; meanwhile, the outcomes of this experiment could be determined.
4. **Verification & Discussion:** This is the most important part in a practicum. In this stage, students have to collect the measured data and then form his/her own estimated knowledge. Thereafter in a panel discussion, this knowledge constructed from the measurement should be compared with the actual law (knowledge) in the real world. And of course, if the error between the estimated knowledge and the real-world law is too large to acceptable, the students then are asked to redo this experiment.
5. **Comprehension Test:** This means that it is necessary to check how much knowledge has been internalized to students through this experiment.

From the above illustration, an *experiment* can be treated formally as an operation that students learn the knowledge (estimate $\hat{K}$ of K) from some manipulations and observations. To obtain more deep analysis of the above processes formally, let the manipulation of this experiment be controlled through some parameters $P = (p_1, p_2, \cdots, p_n)$ and the outcomes of the experiment be constituted as an $O = (o_1, o_2, \cdots, o_n)$.

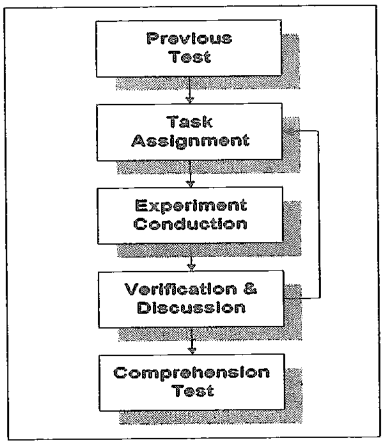

Figure 1 Working Process of Experiment

Experiment model

The outcome O of an experiment is determined by the control parameters P and the nature law in this experiment (actual knowledge K); that is to say,

$$O = O(K, P). \quad (1)$$

On the other hand, the learned knowledge constructed in this experiment is verified by the students through the parameter settings and observed outcomes:

$$\hat{K} = \hat{K}(P, O). \quad (2)$$

The *experiment models* (1) and (2) is described graphically in Figure 2. With this model, the goal and process of an experiment can be easily formulated. Firstly, the goal of an experiment is for students to find out the estimates knowledge $\hat{K}(P, O)$ from the control parameters P and the observed outcomes O. Then, the working process of an experiment can be formulated as follows:

1. **Previous Test:** to decide the background knowledge K_i of the students.
2. **Task Assignment:** to partition the independent variables

$P=(p_1,p_2,\cdots,p_n)$ and dependent variables $O=(o_1,o_2,\cdots,o_n)$ to students for manipulation or observation.

3. **Experiment Conduction:** to execute the nature law (actual knowledge) to produce outcomes $O=O(K,P)$.
4. **Verification & Discussion:** to estimate the experiment knowledge $\hat{K}=\hat{K}(P,O)$ and verify it with variables P and O.
5. **Comprehension Test:** to ensure the learned with actual knowledge K and to determine the knowledge difference $\hat{K}-K_i$ that the students learn in this experiment.

The above model can be used to explain formally the way of knowledge formation in an experiment. In real world, each experiment has its background knowledge (K_i) that we have to learn before experimenting. Then, several formal steps of experiment are processed, such as deciding experiment topic (determining K), preparing the appliance for this experiment (for creating K), setting all the control parameters (P), and then conducting this experiment (producing $O=O(K,P)$). Every time when we get the observation values (O) from environment, we will check the result (verify $\hat{K}=\hat{K}(P,O)$). If it is not correct, we must find out something wrong, and dispel these wrong factors, then we can do it again.

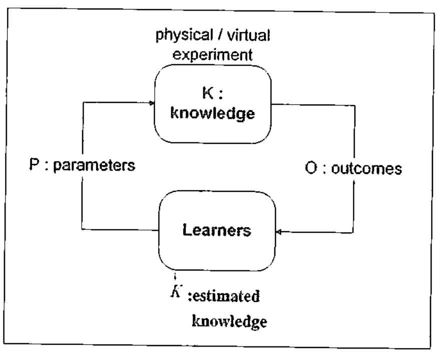

Figure 2. Experiment Model

Simulator and verifier

The two formulae (1) and (2) in the above experiment model can be further analyzed as two state machines with specified inputs and outputs: simulator and verifier. As Figure 3 shown, the simulator is formula (1) $O=O(K,P)$, with inputs being the knowledge embedded in the devised experiment and the parameter assigned the experimenters, and outputs being those outcomes that the experiment would produces. The verifier in Figure 4 implements formula (2) $\hat{K}=\hat{K}(P,O)$ to obtain the estimated knowledge $\hat{K}$ (the outputs) by processing its inputs P and O. In an experiment, the simulator represents the experimenting process that the learners manipulate; whereas the verifier represents the learning process that the learners think.

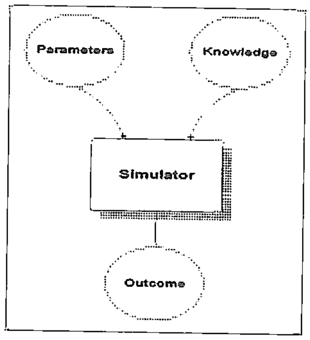

Figure 3. Simulator

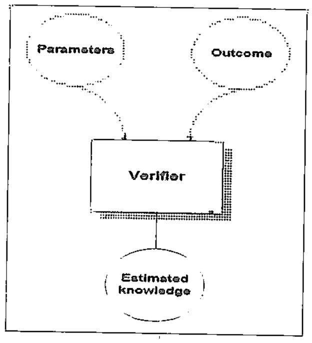

Figure 4. Verifier

Petri-net modeling

Many analyzing tools in system theory can be used to formulate the simulator and verifier in the experiment model. Petri net is chosen here to enhance the transition property of knowledge processing. A Petri net is symbolized as a four-tuple $C=(A,T,U,V)$, with A being the set of *places*, T being the set of *transitions*, U being the input functions and V being the output functions. [5] In the experiment model, all the places are the knowledge K, parameters P, outcomes O and the estimated knowledge $\hat{K}$, and the input and output functions are some collections of these places. The major difference between the simulator and verifier is their transitions. The transitions of the simulator denote the activation of some physical laws to obtain the experimental results (outcomes or intermediate outcomes); whereas, those of the verifier are the knowledge verification steps in the learners' brains, which can learn from the collected data (P and O) to induce the estimated knowledge $\hat{K}$.

For two machines mentioned above, there are two important status: activability and verificableness. If the simulator. can run at a time, it must get enough parameters to evaluate, and this status of getting enough parameters is called *activability*. If it is not activable, some transitions in the simulator. will not be fire and will pass some error messages. The same condition occurs in verifying a specified knowledge. When the learners set the wrong or too few outcomes to verify the knowledge, they are always not able to get the correct formula in this experiment. If the verifier. is set enough outcomes, it meets the condition of *verificableness*. Both activability and verificableness conditions can be studied through the rechability of Petri net [5], which will not investigated further here.

The experiment model achieved causes the two parts of the experiment environment and would be very useful when an experiment is to be generated on computer network virtually. A virtual environment of the practicum is designed with the experiment models in the next section.

4. Design of virtual experiment environment

In this section, we are going to establish a Virtual Experiment Environment (V.E.E. in short) on computer network. Many issues for an experiment mentioned in the previous section, such like the stages of an experiment and the concept of experiment.

Architecture of V.E.E.

The original idea of *V.E.E.* is that students can use it to do experiment through the computer network, so that *virtual environments* can be processed as in a real laboratory. There are two important locations: the first is the experiment room and the second is the equipment storage room. Both rooms are designed at the users' site and compose the whole *client of V.E.E.*. And then, there should be an important part, called *server of V.E.E.*,used to collect some on line data and to obtain some statistics. For this sake, the whole experiment environment including users and experiment objects must be monitored by server, therefore the V.E.E. system is designed as Figure 5.

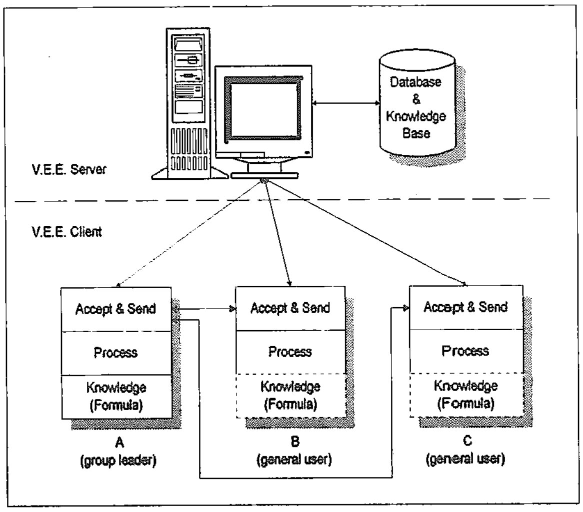

Figure 5. The architecture of V.E.E.

Mission of V.E.E. server is to manage users and collect data, so it must be connected with some database. When a user registers, the server have to give this user an account, and every time he or she login this system, the server should record his/her login time, check his/her privilege, and assign him/her a laboratory to do the experiment he/she wants. Hence, the server must possess a user list, including user account, password, login times, and so on. Besides, the conditions of experiment have to be known by the server, like the duration and the number of correct results of experiments.

V.E.E.'sclient is the virtual environment that users can deal with locally. The V.E.E. allows a group of users to do experiment together to reach the goal of cooperation, so it needs to have a group leader that can control the operations of all in the group. Hence, the user type can be classified into *the leader* and *general users*.

Software components

According to the experiment models proposed in the previous section, we can find out that V.E.E. has two kinds of jobs. One is to run with the parameters setting, and the other is to verify if the formula passed by learners pass into is right. These two parts are the simulator and the verifier, just as Section 3 analyzed.

The simulator is used in the running time, and we design it as that it accepts the input parameters and formula of the experiment and send the outcome that users indicated. When the learners who want to check the formula that had been found out, the verifier will active. The learners reduce the formula according to the parameters they set and the outcomes they get, and then send it to verifier to check if it is correct. Modeling Figure 3 and Figure 4 with Petri net can be used to implement the work of simulator and verifier.

Some visualization and user-defined objects and tools are also included in V.E.E. The objects are simulated according to those in real experiment, and the learners can feel friendly with these objects. The users can order some suitable objects they want by changing the properties of these objects, for example, changing the *weight* property of a ball object to get a 5 Kg. ball. When an experiment is designed, the learners perhaps want to save it for later usage, so V.E.E.is designed to be able to save these experiments. These saved experiments is called *user-define tools*, and every time they are restored, it is still visualized.

The V.E.E. is designed to let students cooperatively do experiment so that it provides some mechanisms to accomplish cooperation. To simulate the condition of real laboratory, an experiment is divided into severalparts to make each user be able to operate some part of the whole experiment work. For example, when three persons are doing an experiment, one of them may decide what object they want, and then the others may arrange position of these objects and record the result. The V.E.E. achieves cooperation in this way.

As the previous thinking, a cooperative dialog has to be defined in V.E.E.for the purpose of cooperation. An experiment is separated into some parts, and each part represents as a dialog that can do a part of job in the experiment. A dialog like this is a cooperative dialog in V.E.E.

V.E.E. operations

Now, let us discuss how many works are in an experiment of V.E.E. According to the working process of an experiment, the learners pass the set parameters into the environment, and get the values of experiment observed in this environment. Besides, these parameters is stored in objects which are used in experiment and have to be selected in advance. Therefore, at least parts of works can be defined. One is selecting objects we need and arranging them in the desk. Another is setting their attributes, such as weight, color and size of the selected objects. The third part needed is to set the values of results to be used and record them. All these are called the *basic works* in V.E.E.

The first step to do a physical experiment is learning how to use experimental equipment, whose counterpart in a computer-simulated visualized environment is the devised visual objects. Take the "Projectile Motion" simulation experiment as an example. After a ball with assigned attributes is placed on the experiment platform, the ball would be thrown to the correct position during experimenting. This ball is called the visual object and its operation method is simulated as a real ball in the real laboratory.

Each experiment is involved with *visual objects* and *knowledge objects*. The former mean those objects visualized in V.E.E.and the latter represent the embedded knowledge to be learned. As Figure 6 shown, the Projectile Motions has to be composed of the knowledge object "Newton's second Law of Motion" and the visual objects Ball, Ground/Wall and Slope in this object hierarchy. Therefore, the experiment Projectile Motions can be performed only after these objects are all available.

In summary, an experiment can be simulated on the computer screen when its process is correctly designated as the Petri nets of the simulator and verifier. The best worth of this V.E.E is that it can simulate those experiments which are difficult or even impossible to be implemented in the real world, such as Kepler's Planet Motion. The next section will describe such an experiment in detail.

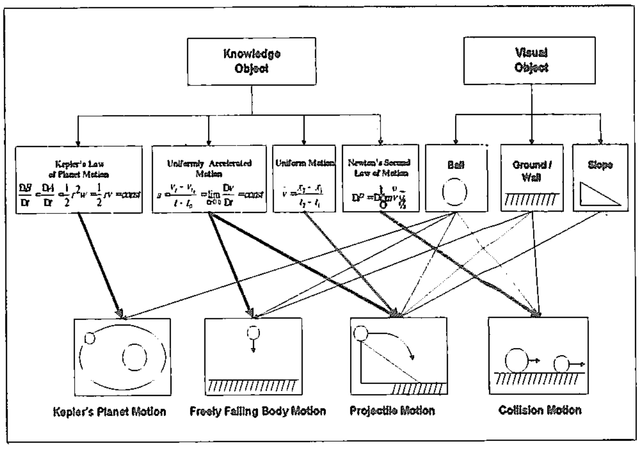

Figure 6. Knowledge and Visual Object Hierarchy

5. Experimental system

The V.E.E. in Figure 5 is developed by Borland Delphi 2.0 on Windows 95 or Windows NT. And the experimental system is described with the Law of Kepler's Planet Motion in this section. Users must login the system in the beginning. If login is successfully, as Figure 7 shown, the system will show a choosing dialog box for experiment mode selecting, users then choose the single user mode or the multi-user mode which them needed. In single user mode, user operates an experiment alone in an individual room. On the contrast, in multi-user mode, each group needs three users, and one of them is leader, called experimental leader.

To be an experimental leader in a group is decided by the login time of the system, that is, the user who choose the multi-user mode to login the V.E.E. earliest is the experimental leader and waits for other partners. There are two ways in order to avoid spending too much time for waiting partners, one is the leader has the right to query the user list of the V.E.E. from server and invite partners to join him/her. Another is the server takes the responsibility on calling partners and dispatching them into groups which have less than three users.

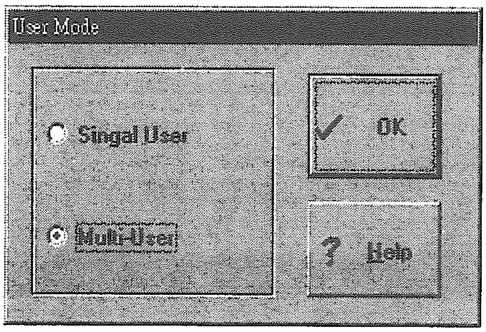

Figure 7. Login stage

After entering the V.E.E. successfully, there is a list of all experiments in the system and users can select an experiment as shown in Figure 8. After finishing selection, a previous test needs to be done by users before taking any experiments as Figure 9 presented. Because some pre-stored knowledge may be needed for the background of the experiment which they chosen, so that the previous test must be passed.

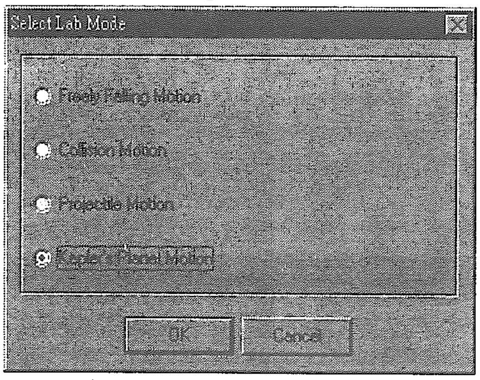

Figure 8. Experiment selected

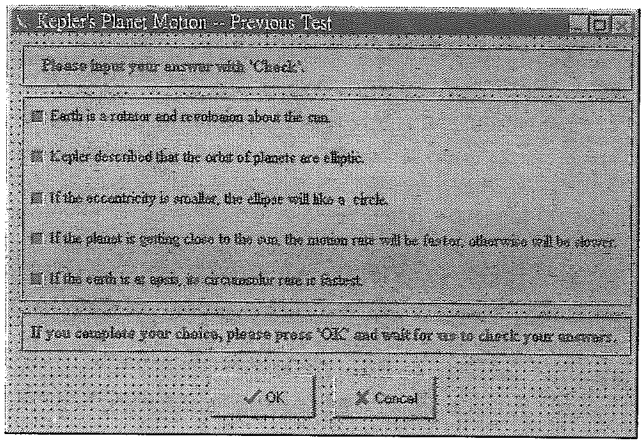

Figure 9. Previous stage

If users who enter withmulti-user mode to the V.E.E, the experimental leader takes the responsibility to dispatch tasks, which includes the experiment equipment arranging, the initial values of the experiment designing, that means to set the properties of Visual Object, and these items of watch form setting is to add items into the "Watch Form".

In the experiment of Kepler's Planet Motion, there are two balls which have been constructed by Visual Ball Object and Kepler's Law of Planet Motion Knowledge Object when choosing the kind of experiment.

Each ball has its own property table and users only need to fill fields of proper value in the table, then the object will execute correctly through the formula which is given by Knowledge Object with its properties. Since the objects of V.E.E. are used to let users design their own experiment scene and prove the correction of them through experimental outcomes, the observant values in "Watch Form" are certainly important. Users must decide the items which needed to obtain correctly based on the sufficient conditions of the formula proving, and then the outcome will be meaningful.

Once dispatching tasks finished, everyone should start their jobs through the experimental features. There is another step called integration before executing the experiment and this step is assumed by experimental leader. The experimental leader must check the property setting of the partners and set the environmental variables because some experiments should set the environmental variables such as the value of "g" in Freely Falling Body Motion experiment. After all, the experimental leader presses execution button to let the environment running with the formula of Knowledge Objects, the properties of Visual Objects and environmental variables, as shown in Figure 10, and the outcomes are also shown in Watch Form.

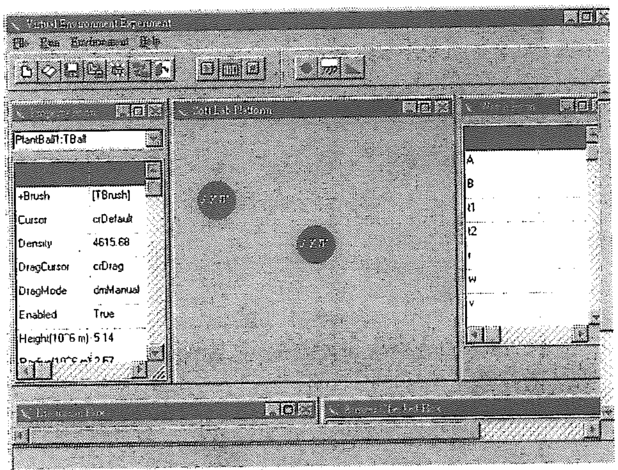

Figure 10. Experiment conduction stage

Following the running stage of experiment, it is the discussion stage, and learners can compute and discuss the experimental results through discussion window, infer the formula from experimental results. Finally, the system does solution check after the results are entered into answer fields. If the answer is not right, repeat the stages from task assignment to verification stage are asked to redo until that the answer which derived by learners is correct. Figure 11 presents the screen layout of verification stage.

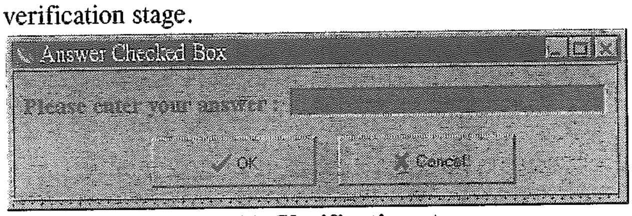

Figure 11. Verification stage

After the verification stage passed, the comprehension test that examine all the knowledge and operating skills of the experiment is coming. This is not only encouraging the memories but avoiding deviant learning result from experiments.

6. Conclusion

Following the architecture of the V.E.E.,the spirit of experiment had been accomplished on the computer network. Learners can learn knowledge by cooperating with others in the virtual laboratory, although they are not at school. Moreover, some theories which are limitedby physical laws in the real world and can not be scheduled into practicum will still possibly be simulated in the V.E.E. to help students operate and learn.

The intelligent companions shall be involved into this V.E.E.to be the tutor of practicum or cooperative learner of users in the future. And based on the experiment model, a full flow of the practicum needed to be analyzed with the Petri-net; besides, a mechanism for controlling the learning movement with the condition of learners should also be the next development direction of the V.E.E..

References

[1] T.W. Chan, "A Tutorial on Social Learning Systems", *Emerging Computer Technologies in Education*

[2] C.J. Huang, "The Curriculum Design", Chinese Edition, Taiwan:Dong Hua Press, 1991

[3] J.S. Heh, W.T. Shu, J.C. Jehng and T.K. Chan, "The Design and Development of a Distributed Multi-agent Visual Learning Environment", *Third International Conference on Intelligent Tutoring Systems*

[4] J.C. Jehng, Y.F. Shih, S. Liang and T.W. Chan, "TurtleGraph: A Computer Supported Cooperative Learning Environment", *Proceeding of World Conference on Educational Multimedia & Hypermedia*, Vancouver, Canada, AACE, pp.292-298

[5] James L. Peterson, "Petri Net Theory and the Modeling of Systems", Prentice-Hall, Inc., Englewood Cliffs, 1981

[6] Sho-Huan Tung, "Visualizing Evaluation in Scheme", 1995