Abstract—Educators continually strive to provide learning materials that are specifically adapted to their students’ unique learning preferences and needs. Software use continues to grow in classrooms however typical educational programs cannot be modified to suit individual learners. The Thinking Head Whiteboard is a lesson authoring tool that provides such a capability, allowing educators without a computer programming background to create their own interactive lessons. The Thinking Head Whiteboard also supports the use of virtual human ‘tutors’ and ‘peers’ within these lessons, making it uniquely suited to assist in areas such as social skills education. Further, the software currently being developed incorporates a basic level of automated assessment, allowing it to adapt on the fly to learner needs, providing repetition of content for struggling learners and fast-tracking those who are proficient. Ultimately, it is hoped that the Thinking Head Whiteboard will become an engaging and useful tool for educators and learners alike.

Keywords—computer aided instruction, autonomous agents, authoring systems.

I. INTRODUCTION

Educational software has long been a feature of many homes and classrooms. More recently, attention has turned to two key areas aimed at enhancing outcomes when using such software – automated assessment and the incorporation of virtual ‘teachers’ and ‘peers’. Automated assessment has been shown to greatly enhance outcomes for struggling students in the areas of literacy and numeracy [1] while the potential of virtual humans for improving skills in those with special needs such as autism and hearing impairments is now being recognized and developed [2,3].

Here we discuss the development of the Thinking Head Whiteboard, software which incorporates both virtual humans and automated assessment technology. In addition, the Whiteboard has been structured in such a way as to allow non-programmers to write their own unique learning tasks and lesson sequences, as well as modify existing materials. We intend for this to become an engaging and effective educational tool that can be utilized in a wide range of learning areas.
the learner to progress [5]. If this is done regularly, the educational process is managed and misunderstandings due to insufficient prior knowledge are minimised, along with repetition of previously mastered content and other difficulties. This philosophy has been adopted within the Thinking Head Whiteboard and while the automated assessment procedure implemented in this software is basic, it is nonetheless continuous, adapts to the user and is structured in such a way that learners have choice, being presented with a small range of tasks appropriate to their current level of knowledge. Providing the learner with choice also helps them to take some control over and ownership of their learning, which has been shown to benefit motivation, and thus educational outcomes [6].

A variety of systems incorporating virtual agents and in-built assessment exist, for example Project LISTEN [7] for reading skills and Wayang Outpost [8] for mathematics. However, while these systems allow basic adjustment of settings and content to suit learners, they do not provide facilities for educators to create their own lessons or significantly alter the ones provided. The Thinking Head Whiteboard not only allows educators to customize existing content, for example replacing provided images with ones better suited to their specific learners, but also to write their own entirely new lessons and curriculum sequences. Further, the virtual human ‘Head X’ is itself customizable, with educators being able to modify its appearance, voice and behaviours to suit the learner.

Educators strive to provide their learners with content that specifically targets their needs and is adapted to their learning style. The Thinking Head Whiteboard not only enables them to do this using an interactive medium, but also allows them to exploit the benefits that both virtual humans and continual automated assessment can bring, providing a self-paced and engaging educational experience that it is hoped will ultimately lead to better educational outcomes for students.

### III. Applications

The Thinking Head Whiteboard is a lesson authoring tool that educators can use to create curricula and activities on any topic they wish. However, there are some applications where access to virtual humans and this self-paced, adaptive approach is anticipated to be particularly advantageous. One such application that is already under development is a social skills tutor for children with autism.

As previously mentioned, virtual humans have been shown to be particularly effective for children with special needs such as hearing and communication impairment or social difficulties. Individuals with autism experience difficulties with language, communication and social skills and can also have atypical sensory tolerances. What this means is that they can become overwhelmed in typical social situations, particularly when multiple individuals are interacting. Social interactions are fast-paced and complex, being made up not only of speech but also face and body gestures. While individuals without autism are generally able to take all of this social information in and process it without conscious effort, those on the autism spectrum often have to be explicitly taught this skill. Virtual humans are ideal for this purpose as they can be programmed to model particular gestures and behaviours and can be used to break down these complex social interactions into their components. This enables scaffolding to occur, where the learner only has extra complexity added when they have fully mastered the earlier, simpler tasks. The Thinking Head Whiteboard is capable of displaying multiple virtual humans simultaneously, making it possible for educators to create activities where social interactions are modeled for the learner and then examined, with key social cues such as facial expressions or hand gestures highlighted and explained, or the reasoning behind individuals’ responses to each other explored.

Using appropriate modeling tools such as FaceGen [9] it is possible to create new faces for use with ‘Head X’. This is relevant to all learners, as having a virtual ‘peer’ that they can relate to, for example one with the same age, gender or cultural background, will assist them to engage with the software [10,11]. Using FaceGen’s ‘PhotoFit’ functionality, it is also possible to create a face that looks like the learner. It is hoped that by replacing one of the socially competent ‘peer’ faces with the face of the learner, the significant learning gains found when using video self-modeling may also be found here [12]. Additionally, the ability to create different faces for the virtual humans means that learners can experience expressions and speech coming from a wide range of individuals. One of the major limitations faced by many interventions for those with autism is a lack of generalization – individuals learn how to perform the intervention task, but fail to perform the same skill in novel situations. It is hoped that by exposing the learner to a wide range of virtual faces, they will learn to recognize the social cues being taught in a more general way, and thus be able to apply that knowledge to actual human faces.

Another application that the Thinking Head Whiteboard system would suit well is akin to the social tutor for autism, and that is a social tutor for individuals with hearing impairments, particularly with a focus on improving communication skills. Individuals with hearing impairment experience challenges developing their social skills, as they often struggle to communicate and engage with their same age peers [13]. Without this interaction, it is difficult to develop social skills at the same rate as a non-hearing impaired child. Using the Thinking Head Whiteboard, virtual humans can be used to model and explicitly draw attention to relevant social cues, allowing for independent practice. Another facet of effective communication is vocabulary, and autonomous virtual humans have already been shown to be effective tools for education in this area. Developed by Bosseler and Massaro, the virtual humans Baldi and Timo have been evaluated both with children suffering from hearing impairments and those on the autism spectrum, with both groups showing improvement immediately following the intervention as well as sustained improvements months later [3]. The Thinking Head Whiteboard could also be used to provide a similar educational experience, with the educator providing vocabulary-focused interactive lessons where an image, video or animation illustrating a target word or phrase is shown and the virtual humans model the speech sounds and facial shapes of that target. Similarly, the learner could type their own words or phrases in and have the virtual humans model it for them. Allowing the hearing impaired learner to explore a new word
or phrase in a variety of ways may help to strengthen long-term recognition and recall of taught content.

It is expected that learners with poor literacy skills will be particularly supported by the Thinking Head Whiteboard approach. Having a virtual human ‘tutor’ to guide them through activities, for example reading instructions aloud and providing hints when requested, means that they can understand what is expected of them and complete tasks independently. This ensures that the learner’s difficulties with reading or writing do not hinder progress in other areas where skills such as comprehension, mathematics or problem solving are the primary focus. The self-paced nature of learning with the system assists all users but especially those with learning difficulties, as these individuals need to be able to repeat similar activities and topics until sufficient mastery is reached. In a classroom context this can be challenging for the teacher to manage logistically, however this software manages the process comfortably due to the in-built automated assessment. While these features are beneficial to all learners and topic areas, they are especially important for those with poor literacy skills and other learning difficulties.

IV. IMPLEMENTATION

A. Architecture Overview

The system discussed here is composed of two standalone programs, the virtual human software ‘Head X’ and the lesson interaction and display software Thinking Head Whiteboard. Initial development of the Whiteboard was undertaken in close consultation with an educator who has an extensive professional background but no experience in software development. This guided ongoing development and assisted us to make the software usable for other educators with similarly limited experience. Lesson files and curriculum sequences are defined using simple XML files which are read in and interpreted by the Thinking Head Whiteboard. Lesson files are displayed alongside the displayed virtual humans as interactive activities, while the curriculum sequences are used to guide the automated assessment system, helping to determine which tasks get presented and when. Educators can modify the provided XML files, for example replacing set images with custom images or photographs that will be more engaging for their particular learner, or can write their own activities entirely.

Within the XML lesson files, educators are able to specify which virtual human should respond to particular events and what they will say or do, for example when a user completes a task correctly the educator may wish to override the default response with something more specific or meaningful for their learner. The Thinking Head Whiteboard reads these commands in from the XML files and then controls the actions of the displayed virtual humans via the Synapse interface, a system that uses memory-mapped files to efficiently synchronize and share data between multiple processes.

User interactions with the system are automatically and continually saved into an XML log file and a summary of the user’s progress through lesson tasks and topics is likewise stored into a XML summary file, an extract of which is shown in Fig. 1. This summary file is used as part of the automated assessment process to determine which tasks to present to the learner next, and can also be viewed and printed by the student using the View Progress function of the Thinking Head Whiteboard, as shown in Fig. 2. It is intended that this be a useful tool for both the learner and the educator, as it enables the learner to self-monitor while also helping to inform the educator of the learner’s current state and their areas of strength and weakness.

B. Virtual Human

The virtual human used here, Head X, can model realistic facial expressions and speech making it ideal for use as a

![Figure 2. Condensed screenshot of the View Progress screen in the Thinking Head Whiteboard](image)

![Figure 3. Head X can model a range of facial expressions including sadness, happiness, anger and surprise](image)
virtual teacher or peer, as demonstrated in Fig. 3. Displaying a single Head X instance is useful, as it is possible to have the virtual human act as a tutor and guide learners through their activities. Additionally, the virtual tutor can explicitly model facial expressions and emotions, which as previously discussed is particularly useful when teaching social skills to children with autism and other social impairments, and can model realistic facial shapes during speech, potentially benefitting children with hearing impairments. It is also possible to display multiple Head X instances simultaneously, enabling educators to write lessons demonstrating interactions between individuals.

The ‘Head X’ software can also be set to display subtitles, allowing for it to be used effectively even if the computer’s sound output is turned off, as is sometimes the case for classroom computers, and catering to learners both with low sensory tolerance for sound and with auditory processing difficulties. For those with low sensory tolerance towards visual input, subtitles can be turned off to simplify the display.

As previously mentioned, Head X can be customized using head models created using external modeling programs such as FaceGen or Blender. For individuals with low tolerance towards visual input, it may be prudent to start with a very basic and less realistic virtual human face and gradually increase the realism and complexity as the student becomes comfortable and confident with the simpler versions. Further, Head X can be customized in terms of the absence or presence and appearance of hair shape and colour and other accessory models including eye glasses and shoulders. All of these can be created in modeling programs such as Blender then placed and controlled via the Head X configuration files. Head X also comes with a range of pre-existing models for hair and accessories, making it possible for educators to create their own unique characters for use with the Thinking Head Whiteboard.

C. Lesson Interaction and Display

The activity display component of the Thinking Head Whiteboard is written primarily in Java, with a small number of libraries written using C++. The first of these libraries enables basic speech recognition functionality to be used within the software, taking advantage of the speech recognition system built into recent Windows operating systems. The second library allows the Java component to control displayed instances of Head X via the Synapse interface as previously mentioned. All resources required for displaying an activity, including the XML curriculum and lesson activity definition files as well as additional resources such as images and videos, are organized in a strictly structured directory hierarchy. This allows educators to easily place and locate resource files and reference them within their XML lesson definition files. Multiple curricula can be supported simultaneously, simply by placing the curriculum file for each in the parent ‘input’ directory, then creating a subdirectory with the same name to contain all the associated XML activity definition files. This allows multiple learning areas to be accessible at once, for example both a literacy program and a social skills program could be loaded and the learner would be able to choose which they wish to focus on in a given session.

1) Lesson Authoring

At present, all lessons must be defined using XML files that follow the specified document type definition provided with the Thinking Head Whiteboard. While this may appear daunting at first for non-programmers, most settings have usable default values so only very simple XML files are necessary for basic functionality, an example of which is shown in Fig. 4. This demonstrates a lesson with a single page, a simple introductory action where the Head X ‘tutor’ speaks to the learner and a basic drag and drop activity. The resulting output is shown in Fig. 5. While this example has only a single page, any number of pages containing any variety of different activity types can be specified. To facilitate simple and convenient creation, each activity type has its own XML category. A list of currently supported activity types can be seen in Table 1. A high level of control over the appearance and behavior of the lesson activity and the virtual humans is possible, including defining what should occur when the learner completes an activity or performs part of a task correctly or incorrectly, though naturally this requires more complex XML lesson definition files to be created by the educator.

2) Lesson Interpretation

Within the Java source code there are a variety of ‘manager’ type classes as well as a ‘display’ class for each activity type described in Table 1. The manager classes are described below:

![Figure 4. A basic XML lesson definition file](image)

![Figure 5. The output of the lesson definition file shown in Fig. 4 above](image)
TABLE I. Lesson Activity Types Available in the Thinking Head Whiteboard Software

<table>
<thead>
<tr>
<th>Activity Type and XML tag(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drag and Drop</td>
<td>Multiple ‘draggable objects’ and ‘drop boxes’ can be defined, suit sorting activities. Both a ‘freestyle’ version as shown in Fig. 5 and a grid version are available.</td>
</tr>
<tr>
<td>Cloze</td>
<td>A ‘fill in the blank’ style activity where a block of text is displayed with parts replaced with empty boxes that the learner must complete. Typed and ‘drag &amp; drop’ input modes are available.</td>
</tr>
<tr>
<td>Question List</td>
<td>One or more questions can be displayed simultaneously, with Likert scale and yes/no style responses both available.</td>
</tr>
<tr>
<td>Flexible Face</td>
<td>A cartoon face that can either be used to display particular expressions, or be interactive with learners trying to achieve a ‘target’ expression by moving control points on the eyes and lips. Multiple faces can also be displayed with yes/no style buttons beneath each face, ideal for sorting activities.</td>
</tr>
<tr>
<td>Highlighter Text</td>
<td>Two modes available. In demonstration mode, the virtual human reads the block of text aloud and certain phrases are highlighted as they are spoken. In interactive mode, the learner must highlight key words or phrases within the block of text with their mouse.</td>
</tr>
<tr>
<td>Concept Map</td>
<td>Multiple nodes and the expected connections between the nodes are defined. The learner must assemble the concept map by dragging nodes into position and clicking from one node to the next to connect them.</td>
</tr>
<tr>
<td>Paint</td>
<td>A basic paint panel allows learners to draw a picture using a variety of brushes and colours then save it.</td>
</tr>
<tr>
<td>Video</td>
<td>Displays a media player within the Thinking Head Whiteboard software. The video controls can be hidden or displayed.</td>
</tr>
<tr>
<td>Browser</td>
<td>Displays a webpage or Flash file from within the Thinking Head Whiteboard software. Can display HTML or Flash from both the ‘resources’ folder and the Internet.</td>
</tr>
<tr>
<td>Social Explorer</td>
<td>A simple game with one user-controlled character. Any number of other characters and items can be defined, and the characters’ emotional states change depending on what interacts are chosen by the user. Mazes for the characters to move through can be automatically generated. Alternatively, obstacles can be manually defined and placed in the XML.</td>
</tr>
<tr>
<td>Other</td>
<td>Static images, labels and buttons can be combined to create custom activity types. Combined with an action tag, buttons become interactive.</td>
</tr>
</tbody>
</table>

**WhiteboardManager**: Main entry point for the program. Builds the Whiteboard frame and contains a helper class to place interactive pages onto the frame and make them visible, performs appropriate actions when the Whiteboard program is closed including saving progress and backing up output.

**InputParser**: Initiates the display of lesson options and when a lesson page is chosen builds the interactive display object to be placed on the Whiteboard frame. This class builds any general components itself, for example the ‘tutor instruction panel’ seen at the bottom of Fig. 5, and delegates activity-specific construction to the appropriate display class. The InputParser also ensures that each page’s introductory action, for example the virtual human reading instructions aloud for the learner, is triggered correctly.

**ActionParser**: Interprets XML ‘action’ elements and triggers the appropriate actions to occur, such as movement between pages of a lesson, initiating virtual human speech, playing sounds and displaying pop ups.

**OutputManager**: Controls the creation of the student, summary and log XML files as well as the reading of and writing to these files, inserting new records in the log each time the Whiteboard is opened or a new lesson is shown, and new records in the summary file when lessons are completed.

**AssessmentManager**: Determines what activity choices to present to the learner next. See 'Automated Assessment' below for a detailed explanation of this process.

**AssessmentViewer**: Builds the ‘View Progress’ display that allows the user to view their progress across lessons and objectives. This can also be printed out, which is useful for educators who wish to keep a record of their learners’ progress.

**HeadManager**: Launches then coordinates up to four virtual humans simultaneously, each represented by its own Head class. The HeadManager controls the positioning of each Head X instance on screen and calls the speech and command methods of the correct Head object, which in turn sends the commands to the corresponding Head X instance.

**XMLParser**: Directly manages reading to and writing from XML input and output files and is used to check the validity of XML input files when the Whiteboard is first loaded. Along with the manager classes there are the display classes that build and control the interactivity of their specific activity (see Table 1). Many of these are general in application, but some are well suited to literacy learning, such as the cloze and highlighter text classes, while others are best suited to social skills learning, such as the social explorer and flexible face. One particularly noteworthy class is the concept map. Many common assessment techniques truly only assess information recall, however evidence suggests that concept maps can assess deeper understanding of a concept [11]. In the Thinking Head Whiteboard, a concept map class has been implemented where educators can define multiple nodes and what to display on those nodes. They can then define which nodes should be connected and whether that connection is directional, for example in a food chain ‘cow ➔ eat ➔ grass’ would be correct but ‘grass ➔ eats ➔ cow’ would not, or non-directional. Finally, educators can define the percentage similarity required between the learner’s concept map and the solution to consider the learner’s map complete. This is useful, as in more complex mind maps multiple correct solutions are often possible.
4. If the prerequisites attribute is missing or contains an empty 'lesson' tag of the lesson XML file itself, as can be seen in Fig. 6, however the prerequisites attribute for this is within the lesson XML file, educators specify core lessons that must be completed by the learner, and can optionally specify 'extra lessons' that can be presented to the learner for additional practice if required. From this, the system determines exactly what activities to present to the learner and in what order, based on lesson prerequisites, previously completed activities and learner proficiency. A curriculum definition can contain multiple topics, with each topic containing one or more objectives. Each objective must contain at least one 'core' lesson, and all objectives must be finished successfully for the topic to be considered complete. Once complete, a lesson will no longer be eligible for inclusion in the list of activity choices presented to the learner, however these activities can be accessed via the 'Repeat Previous Lesson' option on the main menu and repeated as often as desired. A sample curriculum definition file is shown in Fig. 6. As can be seen in Fig. 6, educators can set prerequisites for topics by listing the IDs of other topics in the ‘prerequisites’ attribute of the topic tag. This means that no lessons from the given topic will be displayed to the learner until all earlier topics are completed first, allowing the educator to ensure the learner has gained sufficient prior knowledge before they are exposed to more complex content. Prerequisites can also be set for lessons in the same way, however the prerequisites attribute for this is within the ‘lesson’ tag of the lesson XML file itself, as can be seen in Fig. 4. If the prerequisites attribute is missing or contains an empty string it is assumed that the given topic or lesson has no prerequisites and can be included in the options immediately available to the learner.

Fig. 6 also highlights how a minimum required ‘correctness’ and ‘accuracy’ can be set for each lesson and each objective. The value must be between 1.0 and 0.0, with 1.0 representing 100% accuracy or correctness and 0.0 representing 0%. Correctness here refers to the final state of the activity only, and whether the student has met all of the activity’s requirements. Accuracy gives an indication of how many mistakes the learner has made while completing the task. To calculate accuracy, a tally is kept of both the total number of moves and the number of incorrect moves that the student makes during their interaction with the system. Accuracy is then calculated simply as show in (1).

\[
\text{accuracy} = \frac{\text{total\ moves} - \text{incorrect\ moves}}{\text{total\ moves}} \quad (1)
\]

Consider a ‘drag and drop’ activity such as that illustrated in Fig. 4 and Fig. 5. 100% correctness would be awarded if the learner places all ‘draggable’ objects into their corresponding ‘drop boxes’, with no ‘draggable’ objects in the wrong boxes and any ‘draggable’ objects without a specified ‘drop box’ being left in the blank space. 100% accuracy would be awarded only if the learner completed this task without misplacing any ‘draggable’ object. If the learner placed a ‘draggable’ object into the wrong box then recognised their mistake and moved that object into the correct box, they would still be awarded 100% correctness but their accuracy would be lower.

Setting minimum requirements for lessons and objectives allows educators to specify the level of mastery they require from their learners in order to consider the section complete to a sufficient level and unlock new lessons. An example of this is to have correctness for an activity set to ‘1.0’ and accuracy set to ‘0.8’, meaning that the learner must have the task 100% correct when they submit it, but can have an accuracy of only 80%, i.e. they can make a few mistakes while they are doing the activity provided they fix them before submitting it. If the learner does not reach the minimum required correctness and accuracy, they will be able to repeat the task at a later stage.

The automated assessment function of the Thinking Head Whiteboard endeavours to present all ‘core’ lessons to the learner as quickly as possible while adhering to the set curriculum requirements. To achieve this, the AssessmentManager generates a list of ‘accessible lessons’, that is those with no prerequisites or with all prerequisite tasks already complete, and selects up to three activities for the learner to choose between. The AssessmentManager begins by extracting a list of all core lesson IDs from the curriculum file. Core lessons that are already accessible are added directly to the list, while lessons with incomplete prerequisites are not. For these, the AssessmentManager performs a depth-first search through the lesson’s prerequisite sequence until it finds a lesson that is already unlocked. For example, if my core lesson “ParentLesson” had a prerequisite of “MiddleLesson” and “MiddleLesson” had a prerequisite of “FirstLesson” but “FirstLesson” had no prerequisites, “FirstLesson” would be the ID added to the ‘accessible lessons’ list as it is the first available lesson in the sequence required to unlock the required ‘core’ lesson. While prerequisites can be set for topics as well
as lessons, this process is only applied to lesson selection. With topics, all topics that are both incomplete and unlocked are displayed for the user to choose between.

Only a maximum of three lesson options are provided to the user at any one time. The ‘accessible lessons’ list is typically much longer than this. When lessons are added to the ‘accessible lessons’ list, they are assigned a status – core new, extra new, core previously attempted or extra previously attempted. Priority is first given to new core lessons, followed by previously attempted core lessons, then new extra lessons and finally previously attempted extra lessons. Lessons that have been completed to their minimum required correctness and accuracy are only added to this list if all necessary lessons are complete, but the average correctness or accuracy of all lessons within the objective falls below the minimum requirements. In this case, lessons will require repetition until the objective’s requirements are met. This case depends highly on how the educator structures their curriculum within the XML file.

The approach described here aims to allow learners with more advanced knowledge of the topic being presented to fast-track through and complete only the minimum required lesson sequence, while enabling learners who need more time to access the ‘extra’ lessons to supplement their learning as well as repeat previous activities until their understanding is strong enough to complete the ‘core’ lessons to the required level. While this implementation is a simple example of automated assessment which is strongly driven by the XML curriculum definition file written by educators, it is nonetheless expected to provide a more relevant and engaging educational experience for the learner than can be provided using a static lesson sequence.

V. ACTIVITY AND ASSESSMENT RECOMMENDATIONS

The Thinking Head Whiteboard allows for free form lesson design, however there are some recommendations that educators using the software may like to consider, particularly if developing lessons for children with special needs.

The most basic recommendation is to take advantage of the built in functionality supporting the addition of images and icons. Most elements, from labels and buttons to ‘draggable labels’ in the Drag and Drop activity demonstrated in Fig. 5, support the display of icons as well as text. It is also possible to set an icon for each lesson and topic, as shown in Fig. 7. Making your lessons highly visual not only makes them more appealing, but makes it easier for children with language deficits to understand what their choices are and the meanings behind each element. Without this it can be difficult for these learners to use the software as independently as intended.

Along the same lines, it is highly recommended that instructions are kept concise and use simple language. This benefits all learners as clear instructions reduce confusion and frustration. In particular, individuals with autism can struggle to process and follow complex instructions [14], making this recommendation particularly important for this learner group.

While the presentation of activities is important to ensure both understanding and engagement, assessment tasks should also be carefully chosen. Assessment tasks should match the content being taught to ensure that learner mastery can be adequately determined. Much research supports the use of concept maps for this purpose, and hence they have been included in the Thinking Head Whiteboard activity types. Concept maps are considered to be valuable as they allow learners to demonstrate their understanding of the connections between concepts, rather than simply demonstrating recall of remembered facts [15]. It is recommended that, where possible, topics contain a concept map style assessment or similar, where deeper understanding of the taught content is assessed.

As mentioned previously, the Thinking Head Whiteboard is currently being used to develop a social tutor for children with autism. For this learner group, the ability to generalize learnt skills and knowledge to novel contexts is often a challenge [3,14]. To support these learners in particular, but to the benefit of all learners, it is recommended that a variety of activities and scenarios be presented so that learners can practice the skills being taught in a range of novel contexts. This supports generalization and helps avoid learners associating what they are learning with a single activity type.

The final key suggestion relates to the structure of the curriculum. To keep lesson sequences flexible and able to adapt to both proficient and remedial learners’ needs, it is recommended to create a large number of lessons that are short, concise and focused on a single concept rather than a smaller number of long lessons. While multiple pages can be defined within a single lesson, it is typically suggested that only a single page with a single activity be used. This has the added benefit of creating a sense of progress, as learners can often complete several short lessons in a single sitting. Most people can relate to feeling frustrated and disheartened when a learning task takes them a long time to complete, so this approach assists in keeping learners engaged and motivated.

VI. FUTURE WORK

The Thinking Head Whiteboard is designed to provide a platform that researchers and educators can use for developing and evaluating educational applications that harness the power of both virtual humans and automated assessment for learning. Development of a social skills tutor for children with autism is currently underway and if the outcome of its evaluation is positive, adaptation to other learner groups such as those with hearing impairments may be recommended.

A limitation of the current implementation is that the automatic assessment is very basic. Drawing on research from
the fields of computer aided instruction and computerized adaptive testing, it is intended that a more sophisticated system will replace or be incorporated into the existing one. Existing work has examined the value of probabilistic approaches for identifying gaps in learner knowledge [16,17] and latent semantic analysis for determining how close a learner’s written response is to a provided ‘ideal’ response [18,19]. Information about the learner’s interaction with the system, for example idle time during a task, number of tasks abandoned prior to completion and other data could also be incorporated to give a more holistic view of the learner’s level of concept mastery.

Another limitation of the current system is that lessons and curricula must be defined using XML files. While these have been kept simple and have useable defaults set where possible, this can still be daunting for non-programmers. In future we hope that a visual interface for constructing lesson sequences and curriculum files will be developed.

Future development is also hoped to bring with it some additional ‘pre-built’ activity types as well as increase the sophistication of the existing ones. For example, in the ‘paint’ display, there is currently no way for the software to determine whether the learner has drawn what was asked of them. Being able to ask the child to draw something specific, for example a happy face, and then locate one in their drawing and praise them for it would improve the educational value of that particular activity. Again on the social skills theme, a new activity type where a webcam is utilized as a ‘mirror’, displaying the learner’s face on screen could be used for exploring facial expressions, as learners could be asked to ‘make a happy face’ or copy the virtual human’s displayed expression. Clearly, there are still many interesting activity types that could be added to the Thinking Head Whiteboard.

Since development of the Thinking Head Whiteboard began, a new virtual human technology has become available. The ‘Clevertar’ engine supports the display and animation of a virtual human with a complete body, rather than just the head and shoulders that the current virtual human uses. This would further enhance the educational experience, particularly for social skills, as it is possible for hand and body gestures to be modeled. It is not a task to undertake lightly, however, as significant work will be required to develop both character models and custom animations for these gestures prior to incorporation into the Thinking Head Whiteboard platform.

The Thinking Head Whiteboard is currently undergoing final testing and debugging. The evaluation of the aforementioned social tutor will not only evaluate learning outcomes for that application, but will indicate how well the Whiteboard software performs as a platform. If the outcomes from this are positive, we hope to make the software available to both educators and researchers in the near future.

ACKNOWLEDGMENT

This work was supported in part by the Chinese Natural Science Foundation under Grant No. 61070117 and Grant No. 609730057, and the Beijing Natural Science Foundation under Grant No. 4122004, the Australian Research Council under ARC Thinking Systems Grant No. TS0689874, as well as the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions.

REFERENCES