CarSim: An Automatic 3D Text-to-Scene Conversion System Applied to Road Accident Reports

Ola Åkerberg† **Hans Svensson**† **Bastian Schulz**‡ **Pierre Nugues**†

†Lund University, LTH
Department of Computer science
Box 118, S-221 00 Lund, Sweden
{e94oa, e94hsv}@efd.lth.se
Pierre.Nugues@cs.lth.se

‡Technische Universität Hamburg-Harburg
Schwarzenbergstraße 95
D-21071 Hamburg, Germany
b.schulz@tuhh.de

Abstract

CarSim is an automatic text-to-scene conversion system. It analyzes written descriptions of car accidents and synthesizes 3D scenes of them. The conversion process consists of two stages. An information extraction module creates a tabular description of the accident and a visual simulator generates and animates the scene.

We implemented a first version of CarSim that considered a corpus of texts in French. We redesigned its linguistic modules and its interface and we applied it to texts in English from the National Transportation Safety Board in the United States.

1 Text-to-Scene Conversion

Text-to-scene conversion consists in creating a 2D or 3D geometric description from a natural language text. The resulting scene can be static or animated. To be converted, the text must be appropriate in some sense, that is, contains explicit descriptions of objects and events for which we can form mental images.

Animated 3D graphics have some advantages for the visualization of information. They can reproduce a real scene more accurately and render a sequence of events.

Automatic text-to-scene conversion has been investigated in a few projects. NALIG (Adorni et al., 1984; Di Manzo et al., 1986) is an early system that was designed to recreate static 2D scenes from simple phrases in Italian. WordsEye (Coyne and Sproat, 2001) is a recent and ambitious example. It features a large database of 3D objects that can be animated. CogViSys (Nagel, 2001; Arens et al., 2002) is aimed a visualizing descriptions of simple car maneuvers at crossroads.

All these systems use apparently invented narratives.

2 CarSim

CarSim (Egges et al., 2001; Dupuy et al., 2001) is a program that analyzes texts describing car accidents and visualizes them in a 3D environment. The CarSim architecture consists of two modules. A first module carries out a linguistic analysis of the accident and creates a template – a tabular representation – of the text. A second module creates the 3D scene from the template. The template has been designed so that it contains the information necessary to reproduce and animate the accidents (Figure 1).

A first version of CarSim was designed to process texts in French. We used a corpus of 87 car accident reports written in French and provided by the MAIF insurance company. Texts are short narratives written by one of the drivers after the accident. They correspond to relatively simple accidents: There were no casualties and both drivers agreed on what happened. In spite of this, many reports are pretty complex and sometimes difficult to understand.

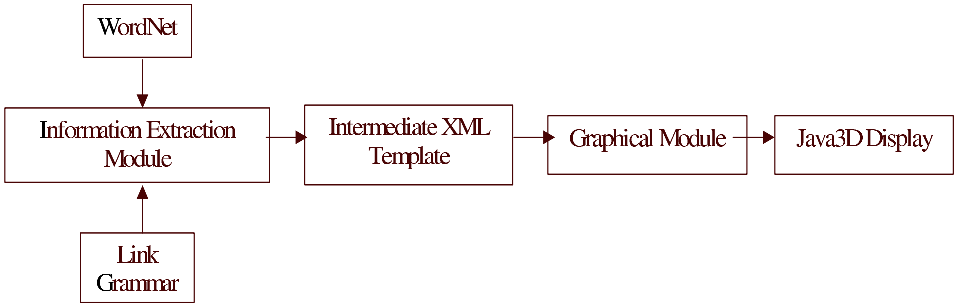


Figure 1: The CarSim architecture.

We describe here a new system that accepts reports in English. We developed and tested it using twenty road accident summaries from the National Transportation Safety Board (www.ntsb.gov), an accident research organization of the United States government. The accidents described by the NTSB are more complex or spectacular than the ones we analyzed in French. To visualize them, we had to add new vehicle actions like "overturn."

3 An Example of Report

The next text is an example of summaries from the NTSB (HAR-00-02):

> *About 10:30 a.m. on October 21, 1999, in Schoharie County, New York, a Kinnicutt Bus Company school bus was transporting 44 students, 5 to 9 years old, and 8 adults on an Albany City School No. 18 field trip. The bus was traveling north on State Route 30A as it approached the intersection with State Route 7, which is about 1.5 miles east of Central Bridge, New York. Concurrently, an MVF Construction Company dump truck, towing a utility trailer, was traveling west on State Route 7. The dump truck was occupied by the driver and a passenger. As the bus approached the intersection, it failed to stop as required and was struck by the dump truck. Seven bus passengers sustained serious injuries, 28 bus passengers and the truckdriver received minor injuries. Thirteen bus passengers, the busdriver, and the truck passenger were uninjured.*

This text is a good example of the possible content of the NTSB summaries. It describes a bus driving on State Route 30A and a truck on State Route 7 and their accident in an intersection. Although the interaction is visually simple, the text is rather difficult to understand because of the profusion of details.

We believe that the conversion of a text to a scene can help understand its information content as it can make it more concrete to a user. Although we don't claim that a sequence of images can replace a text, we are sure that it can complement it. And automatic conversion techniques can make this process faster and easier.

4 The Language Processing Module

The CarSim language processing module uses information extraction techniques to fill a template from the accident narrative. The information extracted from the text is mapped onto a predefined XML structure that consists of three parts: the static objects, the dynamic objects, and the collision objects. The static objects are the non-moving objects such as trees, obstacles, and road signs. The dynamic objects are moving objects, the vehicles. Examples of dynamic objects are cars and trucks. The collision object structure describes the interaction between dynamic objects and/or static objects.

We used two available linguistic resources to analyze the texts: the WordNet lexical database (Fellbaum, 1998) and the Link Grammar dependency parser (Sleator and Temperley, 1993). The strategy to determine the accidents and the actors is to find the collision verbs. CarSim uses regular expressions to search verb patterns in texts. Then, CarSim extracts the dependents of the verb. It evaluates the grammatical function of the word groups, examines words, classifies them using the WordNet hierarchy, and fills the XML template (Åkerberg and Svensson, 2002). Table 1 shows the template corresponding to text HAR-00-02.

Table 1: The template representing the text HAR-00-02 from the NTSB.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE accident SYSTEM "accident.dtd">
<accident>
  <staticObjects>
    <road kind="crossroads"/>
  </staticObjects>
  <dynamicObjects>
    <vehicle id="bus1" kind="truck"
     initDirection="north">
      <startSign>Route 30A</startSign>
      <eventChain>
        <event kind="driving_forward"/>
      </eventChain>
    </vehicle>
    <vehicle id="truck2" kind="truck"
     initDirection="west">
      <startSign>State Route 7</startSign>
      <eventChain>
        <event kind="driving_forward"/>
      </eventChain>
    </vehicle>
  </dynamicObjects>
  <collisions>
    <collision>
      <actor id="bus1" side="unknown"/>
      <victim id="truck2" side="unknown"/>
    </collision>
  </collisions>
</accident>
```

5 The Visualization Module

The visualizer reads its input from the template description. It synthesizes a symbolic 3D scene and animates the vehicles (Egges et al., 2001). The scene generation algorithm positions the static objects and plans the vehicle motions. It uses inference rules to check the consistency of the template description and to estimate the 3D start and end coordinates of the vehicles.

The visualizer uses a planner to generate the vehicle trajectories. A first stage determines the start and end positions of the vehicles from the initial directions, the configuration of the other objects in the scene, and the chain of events as if they were no accident. Then, a second stage alters these trajectories to insert the collisions according to the accident slots in the template. Figure 2 shows the visual output corresponding to text HAR-00-02.

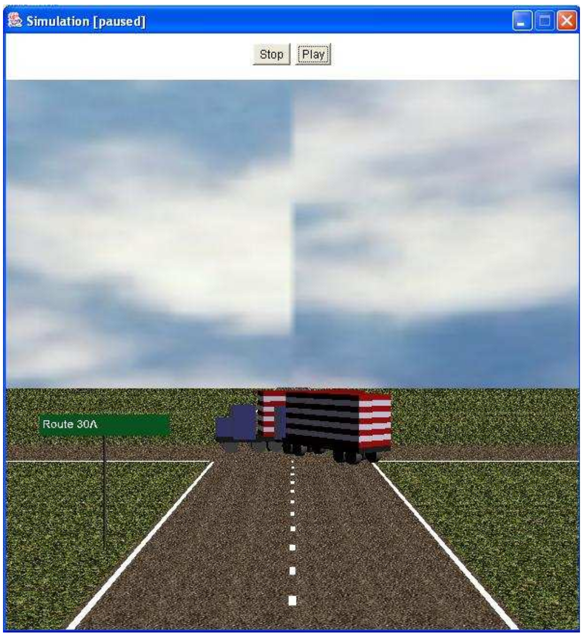
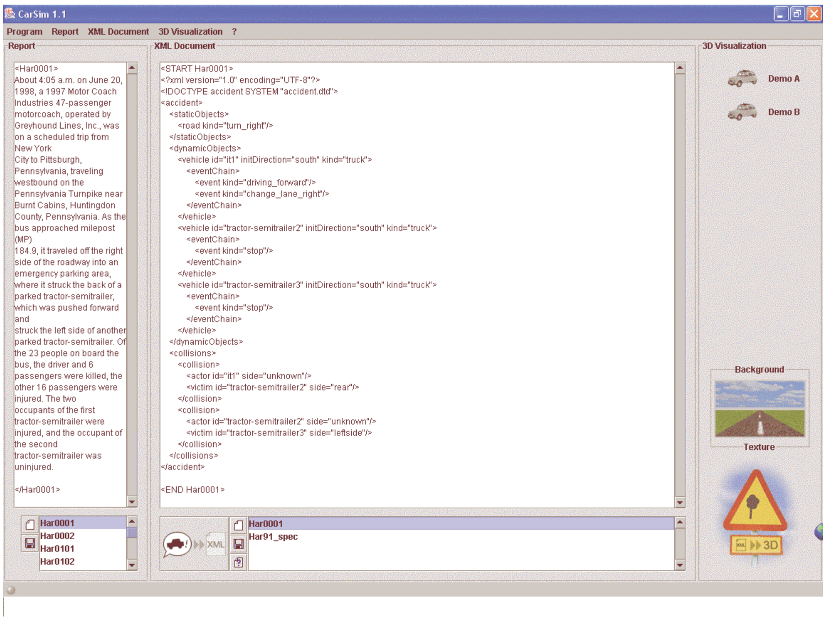


Figure 2: Generated scene corresponding to text HAR-00-02 of the NTSB.

The information extraction and visualization modules are both written in Java. They use JNI as an interface with the external C libraries. All the modules are integrated in a same graphical user interface (Figure 3). The interface is designed to represent text-to-scene processing flow. The left pane contains the original text. The middle pane contains the XML template, and the 3D animation is displayed in a floating window (Schulz, 2002). The interface supports direct editing of the original text file and the XML template. The user can launch the information extraction and the three dimensional simulation of an accident using the bottom buttons. S/he can also adjust the settings of the program.

As far as we know, CarSim is the only text-to-scene converter that is applied to non-invented narratives.



```
<Har0001>
About 4:05 a.m. on June 20,
1998, a 1997 Motor Coach
Industries 47-passenger
motorcoach, operated by
Greyhound Lines, Inc., was
on a scheduled trip from
New York
City to Pittsburgh,
Pennsylvania, traveling
westbound on the
Pennsylvania Turnpike near
Burnt Cabins, Huntingdon
County, Pennsylvania. As the
bus approached milepost
(MP)
184.9, it traveled off the right
side of the roadway into an
emergency parking area,
where it struck the back of a
parked tractor-semitrailer,
which was pushed forward
and
struck the left side of another
parked tractor-semitrailer. Of
the 23 people on board the
bus, the driver and 6
passengers were killed, the
other 16 passengers were
injured. The two
occupants of the first
tractor-semitrailer were
injured, and the occupant of
the second
tractor-semitrailer was
uninjured.

</Har0001>
```

```
<START Har0001>
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE accident SYSTEM "accident.dtd">
<accident>
  <staticObjects>
    <road kind="turn_right"/>
  </staticObjects>
  <dynamicObjects>
    <vehicle id="it1" initDirection="south" kind="truck">
      <eventChain>
        <event kind="driving_forward"/>
        <event kind="change_lane_right"/>
      </eventChain>
    </vehicle>
    <vehicle id="tractor-semitrailer2" initDirection="south" kind="truck">
      <eventChain>
        <event kind="stop"/>
      </eventChain>
    </vehicle>
    <vehicle id="tractor-semitrailer3" initDirection="south" kind="truck">
      <eventChain>
        <event kind="stop"/>
      </eventChain>
    </vehicle>
  </dynamicObjects>
  <collisions>
    <collision>
      <actor id="it1" side="unknown"/>
      <victim id="tractor-semitrailer2" side="rear"/>
    </collision>
    <collision>
      <actor id="tractor-semitrailer2" side="unknown"/>
      <victim id="tractor-semitrailer3" side="leftside"/>
    </collision>
  </collisions>
</accident>

<END Har0001>
```

Figure 3: The CarSim graphical user interface.

Acknowledgments

This work is partly supported by grant number 2002-02380 from the Vinnova Språkteknologi program.

References

Giovanni Adorni, Mauro Di Manzo, and Fausto Giunchiglia. 1984. Natural language driven image generation. In *Proceedings of COLING 84*, pages 495–500, Stanford, California.

Michael Arens, Artur Ottlik, and Hans-Hellmut Nagel. 2002. Natural language texts for a cognitive vision system. In Frank van Harmelen, editor, *ECAI2002, Proceedings of the 15th European Conference on Artificial Intelligence*, Lyon, July 21-26.

Bob Coyne and Richard Sproat. 2001. Wordseye: An automatic text-to-scene conversion system. In *Proceedings of the Siggraph Conference*, Los Angeles.

Sylvain Dupuy, Arjan Egges, Vincent Legendre, and Pierre Nugues. 2001. Generating a 3D simulation of a car accident from a written description in natural language: The Carsim system. In *Proceedings of The Workshop on Temporal and Spatial Information Processing*, pages 1–8, Toulouse, July 7. Association for Computational Linguistics.

Arjan Egges, Anton Nijholt, and Pierre Nugues. 2001. Generating a 3D simulation of a car accident from a formal description. In Venetia Giagourta and Michael G. Strintzis, editors, *Proceedings of The International Conference on Augmented, Virtual Environments and Three-Dimensional Imaging (ICAV3D)*, pages 220–223, Mykonos, Greece, May 30-June 01.

Christiane Fellbaum, editor. 1998. *WordNet: An electronic lexical database*. MIT Press.

Mauro Di Manzo, Giovanni Adorni, and Fausto Giunchiglia. 1986. Reasoning about scene descriptions. *IEEE Proceedings – Special Issue on Natural Language*, 74(7):1013–1025.

Hans-Hellmut Nagel. 2001. Toward a cognitive vision system. Technical report, Universität Karlsruhe (TH), http://kogs.iaks.uni-karlsruhe.de/CogViSys.

Ola Åkerberg and Hans Svensson. 2002. Development and integration of linguistic components for an automatic text-to-scene conversion system. Master's thesis, Lunds universitet, Sweden.

Bastian Schulz. 2002. Development of an interface and visualization components for a text-to-scene converter. Master's thesis, Lunds universitet, Sweden.

Daniel Sleator and Davy Temperley. 1993. Parsing English with a link grammar. In *Third International Workshop on Parsing Technologies*, Tilburg, The Netherlands, August.