Supplemental Material: A Natural Language Interface with Casual Users for Crowd Animation

ABSTRACT

This article contains the results of the user study, some examples, and information about dependency structures of sentences.

1 USER STUDY

There were twelve participants in the user study. They filled three questionnaires, including SUS, background questionnaire, and system evaluation questionnaire. The questions in the last two questionnaires were in 5-point Likert scale (1= not very good/not very agree, 3 = average/neutral, 5 = very good/very agree).

The average score of System Usability Scale was 72.5 with SD = 12.10, indicating that the system was useful in producing crowd animation. Table 1 shows the questions of the questionnaire for the participants' background information. Table 2 shows the questionnaire of evaluating the tools of our system. There were 13 questions. The average score was 4.1 of all the questions. Most of the participants agreed that they were able to produce a crowd animation shortly. They agreed that the characters of crowds were created at desired locations.

Table 1: Questionnaire for user background

No	Question	Score
1	How good is your written English?	3.17
2	How good is your experience in producing animations ?	2.67
3	How good is your experience in producing crowd animations ?	2.00

Our system could perform an automatic way to compute the activation times of the basic animation units based on some preconditions (in Animation Adjustment). We divided the 12 participants equally into two groups A and B to evaluate the usability of this mechanism. There are two conditions: Without automatic mechanism (AM/no) and With automatic mechanism (AM/yes).

Group A: The participants first created the crowd animation under the Condition AM/no and then performed the task one more time under the Condition AM/yes. Group B: The participants of B performed the tasks in the reverse order. They created the crowd animation under the Condition AM/yes and then under the Condition AM/no.

Table 3 shows that the impact was significant on the production time ($p < 0.05$) and the number of attempts in animation adjustment ($p < 0.05$) by the participants between the two conditions AM/no and AM/yes.

2 THE USER INTERFACE AND EXAMPLES

Figure 1 shows the user interface consisting of four components. The components are input component, visual feedback component, crowd scene component, and editing component.

Table 2: Questionnaire for system evaluation

No	Question	Score
1	Do you agree that our system is intuitive to use to produce crowd animations?	4.00
2	Do you agree that our system generates crowd animations naturally?	3.92
3	Do you agree that interactions between crowds are natural?	3.50
4	Do you agree that our system can produce crowd animations that are similar to real crowd video?	4.25
5	Do you agree that the crowd animation produced using our system is similar to the desired animation?	3.75
6	Do you agree that our system generates characters at the desired locations?	4.25
7	Do you agree that our system animates the behaviors of the characters at the desired locations?	3.83
8	Do you agree that the crowd quantity produced using our system is close to the desired quantity ?	3.75
9	Do you agree that the automatic arrangement of crowd events are useful?	4.58
10	Do you agree that the timeline adjustment tool is useful?	4.33
11	Do you agree that animation editing tools are useful?	4.33
12	Do you agree that the results display of the user interface is useful?	4.50
13	Do you agree that the information extracted from the sentence in our system is sufficient to produce desired animation?	4.25

The input component has an input field. The input field accepts sentences. The visual feedback component shows the parsed results obtained from sentence analysis. It also shows the region of crowd location and destination, and the suggested number of character quantity. The quantity can be also adjusted by users.

The crowd scene component renders and animates characters of crowds in real-time. The editing mode and the viewing mode can be switched. After a new sentence is input, the corresponding character model outlines are displayed in green, so that users can observe the new changes.

The editing component allows users to manipulate activation time of each BAU along the timeline, and animation flow of BAUs. An existing BAU can also be deleted. Users can specify the preconditions of BAUs in the animation flow. While the animation is produced, we store the attributes of all the characters. Thus users can select a certain time instance to view the intermediate result. If a sentence is changed, we have to reproduce the animation result. Figure 2 shows examples which were produced by participants.

3 DEPENDENCY RELATIONS

The following briefly describes the dependency relations between words in a sentence:

		Mean	S.D.	Paired Differences		Sig. (2-tailed)
				Mean	S.D.	
Production Time (min.)	No automatic activation time	15.04	5.48	3.42	5.48	0.02
	Automatic activation time	11.63	3.58			
Timeline Adjustment (times)	No automatic activation time	8.25	5.70	5.58	3.58	0.01
	Automatic activation time	2.67	5.70			

Table 3: Results for with and without automatic activation time computation

Figure 1: The user interface. The input component accepts strings as input. It is on top left. The visual feedback component is on the top right. It shows the parsed result of a sentence. In this case, we have quantity (*117*), characters (*people*), and behavior (*walk*). The crowd scene component renders and animates characters of crowds in real-time. The characters are highlighted in green when they are being controlled or created so that users can view the result at once and the amount of the characters can be adjusted promptly. The editing component is on the bottom. The basic animation units are linked and each of them is represented by an icon with a behavior label. The behavior label is useful when the basic animation units are manipulated. A basic unit can be deleted and our system regenerates the crowd scene accordingly. Scene events (e.g., Event 1 and Event 2) can be set as a precondition of a basic animation unit. The arrows indicate the execution order of the basic animation units. A time line is at the bottom-most part of the user interface. Users can drag a play time at an instance to start the animation.

(1) root: A fake node "Root" is used as the governor.
(2) amod: Adjectival modifier.
(3) case: The case relation is used for any case-marking element which is treated as a separate syntactic word (including prepositions, postpositions, and clitic case markers).

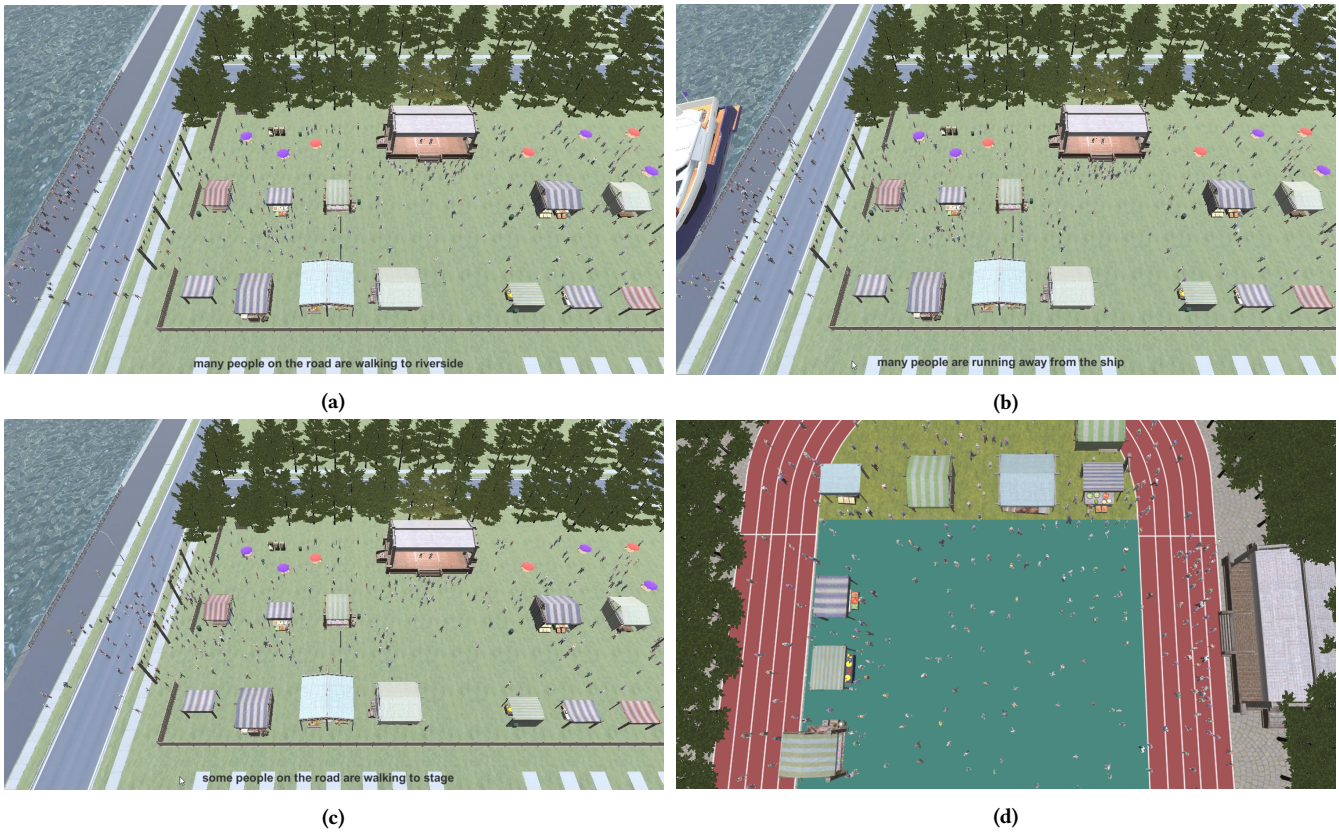

Figure 2: Results produced by participants in the user study. Participant P9 created (a) (b) (c) in Task 1. (a) *Many people on the road are walking to the riverside.* (b) *Many people are running away from the ship.* (c) *Some people on the road are walking to stage.* (d) Participant P6 created the scene. It was produced based on a video of a crowd in the real environment. The scene model was created at the pre-processing stage.

(4) det: The relation determiner (det) holds between a nominal head and its determiner.
(5) nsubj: A nominal subject is a noun phrase which is the syntactic subject of a clause.
(6) nummod: A numeric modifier of a noun is a number phrase that serves to modify the meaning of the noun with a quantity.

Further explanations and examples can be found in Stanford typed dependencies manual [1] and [2].

REFERENCES

[1] Marie-Catherine De Marneffe and Christopher D Manning. 2008. *Stanford typed dependencies manual.* Technical Report. Technical report, Stanford University.
[2] Sebastian Schuster and Christopher D. Manning. 2016. *Universal Dependencies documentation.* https://universaldependencies.org/