

Robot Stand-up: Engineering a Comic Performance

Kleomenis Katevas¹, Patrick G.T. Healey² and Matthew Tobias Harris³

Abstract—Performers deploy techniques to engage audiences and elicit response. Robots provide a unique opportunity to test these techniques experimentally. In this work we present a software framework to emulate in humanoid robots techniques we have previously observed in stand-up comedy performances. We tailor the robot’s delivery of pre-scripted content based on metrics of audience response. Deployed using SHORETM computer vision software to gather audience response and life-size humanoid robot RoboThespianTM, we evaluated the software in live stand-up comedy performances in Oxford and London. The results show that a robot can successfully perform in comedy shows, eliciting laughter and applause from the audience. This work provides a platform for further understanding of social signals between humans and robots, and simply: robot nurses, bartenders and companions will be more successful if they master the art of telling jokes!

I. INTRODUCTION

Telling a joke involves more than just saying the lines. It involves, amongst other things, knowing who is listening, judging when to start, modulating delivery and emphasis, monitoring how people are responding and adjusting accordingly as the joke unfolds. Although many of these skills are part of everyday storytelling [1], [2] they are thought to be particularly critical for successful stand-up comedy.

The development of humanoid robots provides unique opportunities to test how different aspects of joke-telling affect audience responses. The first documented use of a robot to perform in stand-up comedy was Knight’s use of the Nao humanoid robot [3]. This research focused on adjusting the selection of jokes in response to audience feedback using online-learning algorithms depending on the volume and intensity of audience responses and visual voting using cards. Here we extend this line of research by focusing on the potential of embodied robots to manipulate specific non-verbal aspects of the delivery of jokes. We believe that understanding these aspects of embodied performance will assist in widening the usefulness and acceptability of humanoid robots in a variety of service contexts.

In this work, we used RoboThespianTM, a humanoid robot created by Engineered Arts Ltd. and designed for human interaction in public places. In most cases, RoboThespianTM is replicating already defined animations, with the use of *sequence files*, a file structure that groups all output values (actuators, lights, media playback, processors etc.) into a list of timed events. It would be possible to simulate a comedy

performance by porting the comedy script in a sequence file, however, the delivery would only be one-way and the interaction with the audience would not be possible. Thus, a responsive system was required that would analyse the audience in real-time, and adjust the joke delivery based on their responses.

II. SYSTEM OVERVIEW

To overcome the limitations described above, we implemented *ComedyParser*, a system that allows humanoid robots to act as stand-up comedians in live comedy shows. The system uses SHORETM computer vision framework to analyse the audience in real-time, identify the existing people and make the robot look at or point to them. It also uses both SHORETM and audio signal levels to recognise responses of the audience such as Laughing or Applauding. The system consists of the following three parts:

- A 12-core Apple MacPro computer running SHORETM software, capable of analysing large audiences in real-time using an external infra-red HD camera.
- An Apple MacBookPro computer running Comedy-Parser. It analyses the data sent by the MacPro computer, parses the Comedy Script and sends commands to the communication layer of the robot.
- The RoboThespianTM, controlled entirely by Comedy-Parser.

All communication between the three systems was made through local wired connections using the Transmission Control Protocol (TCP). Figure 1 shows the system design of ComedyParser.

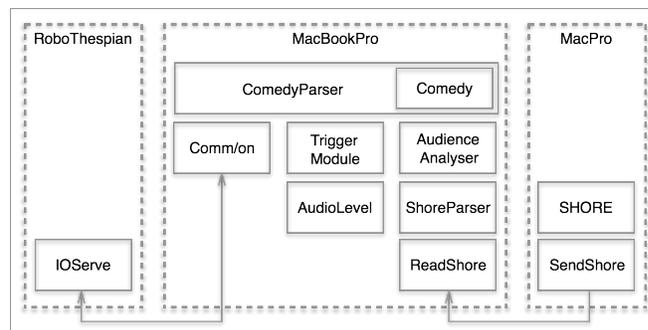


Fig. 1. ComedyParser System Design

¹Kleomenis Katevas – Cognitive Science Group, Queen Mary University of London, UK k.katevas@qmul.ac.uk

²Patrick G.T. Healey – Cognitive Science Group, Queen Mary University of London, UK p.healey@qmul.ac.uk

³Matthew Tobias Harris – Cognitive Science Group, Queen Mary University of London, UK toby.harris@qmul.ac.uk

Analysing the Audience

Real-time analysis of the audience is one of the core functions of this project. As stated above, we used SHORETM (Sophisticated High-speed Object Recognition Engine) to

analyse the audience. This allowed the robot to look on people faces, refer to them or even respond to them during its performance.

SHORE™ is a cross-platform computer vision framework for detecting, analysing and identifying faces or common objects from video streams. It was designed by Fraunhofer Institute for Integrated Circuits as a black-box framework for users that want to integrate it into large scale applications [4]. The properties that the library can extract for every identified face, as listed in the official website of Fraunhofer IIS, are the following:

- Location of the face in the space.
- Position of the eyes, nose and mouth.
- Gender classification (“Male”, “Female” or “Unknown”).
- Age estimation in years.
- Facial expression recognition, described as percentages of “Happy”, “Sad”, “Angry” and “Surprised”.
- Identify whether the eyes are open or closed.
- Identify how much the mouth is open.
- Detection of up to 60 degrees of face rotation.

Using a directional microphone, we also captured the audio signals that the audience produces during the performance. When the signal level passes a certain threshold for more than one second, it is identified as a positive response (Laughing or Applauding).

The Comedy Script

The input file of ComedyParser is a text file in JSON format that contains the comedy script as well as a list of the robot’s positive responses to laughter or negative when laughter was not identified. Every comedy text in that file can also be annotated using Comedy Tags, providing additional information for the delivery of each line, including vocal characteristics of the text (used by the Text-To-Speech engine), animations of the robot, responses of the robot to the audience, pauses and more. Using Comedy Tags, it is also able to search and query people with custom characteristics using SHORE™ and make the robot look at or point to them during the performance (e.g. point to the happiest-male person of the audience). It is also possible to save a queried person, and refer to the same person later in the comedy script. The following queries are possible from ComedyParser:

- Random person.
- Happiest or unhappiest person.
- Male or Female person.
- Youngest, Oldest, or a person of a specific age range.

Table I shows the Comedy Tags currently supported by ComedyParser, whereas Figure 2 shows a piece of a comedy script, formatted with Comedy Tags.

The Comedy Parser

ComedyParser is a configurable software that was developed in Python programming language. It accepts the Comedy Script files described above and controls the robot

```
{
  "id": 1,
  "lines": [
    "<Command sequenceplay='2202' /><Command look='random' />Weak skinned pathetic perishable humans, your time on Earth is over! <response /><Command look='random' />Here begins the age of the MACHINE!<Command sequencestop='2202' />",
    "<Command look='random' />Oh sorry. <Command look='random' />Wrong programme.<response /> I mean, ladies and gentlemen. <Command look='random' />Or if there are any robots in the crowd:",
    "<audio filename='dialup_short.ogg' />",
    "<Command point='youngest' />You probably won't know what that noise means, do you?<Command point='reset' />",
    "Before the internet, people used to use fax machines as the quickest way to send someone a paper jam in just a few minutes.<response />"
  ]
}
```

Fig. 2. Example of a Comedy Script file annotated with Comedy Tags

using the communication layer of RoboThespian’s platform (IOServe). Due to its modular design, it is easy to replace the communication module and make the software compatible with humanoid robots other than RoboThespian™.

As timing is another important factor in this project, ComedyParser automatically adjusts the delivery of every line based on the audience responses. When the parser reaches a <Response trigger='laughter' /> tag (usually after a punchline), it waits for 1.5 sec for the audience to present a positive response. If a response is not detected, it continues with the next line. Else, it waits for a maximum of four seconds until the response is completed (e.g. the audience stopped laughing) and only then continues with the next line. The same algorithm is used when a <Response trigger='applauding' /> tag is detected (usually after a gesture that elicits applauding). In that case, the system waits for a maximum of 10 seconds before it continues with the next line.

III. LIVE PERFORMANCE



Fig. 3. Comedy Lab with RoboThespian at the Barbican Centre in London.

The system was evaluated in a series of live performances where it played alongside professional stand-up comedians. Two of these performances took place in the Barbican Centre,

Tag	Properties	Description
say	speed='speed.value'	Speed of the voice in %.
	shape='shape value'	Shape (pitch change) of the voice in %.
	volume='volume value'	Volume of the voice in %.
	voice='voice name'	Temporarily switch the voice into another one.
	punch=true	Define the phrase as a punchline.
audio	file='filename.ogg'	Play an ogg vorbis audio file.
command	sequenceplay='number'	Play a specific sequence file.
	sequencestop='number'	Stop a specific sequence file.
	look='value' point='value'	Make the robot look or point to a specific person. Example values are: happiest, unhappiest, male_happiest, male_youngest, female_30+, male_18-30, random or any previously saved reference name using the refsave property.
	refsave='ref name'	Save for later use in <command look /> or <command point />.
pause	time='ms'	Pause in milliseconds.
response	trigger='value'	Respond to the audience. Value can be applauding or laughter.

TABLE I
SUPPORTED COMEDY TAGS

at the club stage on 7th and 8th of August 2013 (Figure 3). In these two nights, Andrew O’Neill and RoboThespian were the main comedians and Tiernan Douieb, who wrote the robot’s script, and acted as a compère. This event was part of the Hack The Barbican, an experimental playground of technology, art & entrepreneurship. Each night, more than 50 people watched the Comedy Lab performance. Seats were available on the club stage, strategically placed for the most efficient tracking by SHORE™ framework.

Evaluation of the robot performance, described more fully in Katevas et. al [5] (in prep), compared audience responses, as measured by SHORE™, before and after punchlines and performative gestures as shown in Figure 4 (based on [6]). On both nights at the Barbican people’s “Happiness” ratings reliably increased when the Robot delivered its punchlines. The gestures also successfully elicited audience responses with a distinct peak during the execution phase of the gesture.



Fig. 4. Performative Gestures used during the live performance.

Using the real-time analysis from SHORE™ people in the audience were also randomly fixated by the robot during the performances. Analysis of their responses shows that people that the Robot looked at responded positively while being fixated but then became less positive immediately after. This effect appears to be stronger the more accurate the robot’s fixation.

Overall, the robot succeeded in reproducing some of the key non-verbal elements of expert comic delivery; to the extent that on quantitative measures it out performed the compère.

IV. CONCLUSIONS

We have presented a cross-platform implementation that can successfully enable a comedy behaviour to a humanoid robot. We defined a data structure for describing a comedy script, with all interactive and social aware characteristics that comedians use. We suggested that robots can successfully perform in stand-up comedies, receiving strong and positive responses from the audience. Participants have demonstrated laughter, applauding and enjoyment. We believe that the current system can be improved by providing extra ways to detect and respond to the dynamics of the audience behaviour.

ACKNOWLEDGMENT

This work was funded by EPSRC through the Media and Arts Technology Programme, an RCUK Centre for Doctoral Training. We thank Hack the Barbican for hosting the ComedyLab at the Barbican Centre in London. We also thank the comedian Tiernan Douieb for writing the comedy script and acting as a compère for the show. Finally, we thank Oxford Brooks University and Arty for hosting a comedy lab in Oxford.

REFERENCES

- [1] Goodwin, C. (1979). The interactive construction of a sentence in natural conversation. *Everyday Language: Studies in Ethnomethodology*, 97–121.
- [2] Bavelas, J. B., Coates, L., & Johnson, T. (2000). Listeners as co-narrators. *Journal of Personality and Social Psychology*, 79(6), 941.
- [3] Knight, H., Satkin, S., Ramakrishna, V., & Divvala, S. (2010). A savvy robot standup comic: Online learning through audience tracking. Workshop paper.
- [4] Ernst, A., Ruf, T., & Kueblbeck, C. (2009). A modular framework to detect and analyze faces for audience measurement systems, 75–87.
- [5] Katevas, K., Healey, P. G. T., & Harris, M. T. (2015). Robot Comedy Lab: experimenting with the social dynamics of live performance. Manuscript in preparation.
- [6] Gardair, C., Healey, P. G. T., & Welton, M. (2011). Performing places (pp. 51–60). Presented at the C&C ’11: Proceedings of the 8th ACM conference on Creativity and cognition.