Facial Expression Synthesis on Robots: An ROS Module

Maryam Moosaei, Cory J. Hayes, and Laurel D. Riek
Dept. of Computer Science, University of Notre Dame
{mmoosaei,chayes3,lriek}@nd.edu

ABSTRACT

We present a generalized technique for easily synthesizing facial expressions on robotic faces. In contrast to other work, our approach works in near real time with a high level of accuracy, does not require any manual labeling, is a fully open-source ROS module, and can enable the research community to perform objective and systematic comparisons between the expressive capabilities of different robots.

Categories and Subject Descriptors: {Computer systems organization} {Robotic control}

1. INTRODUCTION

In HRI, we would one day like robots to be capable partners, able to perform tasks independently and effectively communicate their intentions. While some robots are highly dexterous and can express intention through a wide range of body movements, one noticeable limitation is that there are numerous subtle cognitive states, like confusion, disorientation, and attention, which cannot be easily expressed without a face.

The human face is a rich spontaneous channel for the communication of social and emotional displays. Facial expressions can be used to enhance conversation, show empathy, and acknowledge the actions of others. Thus, robot behavior that includes at least some rudimentary, human-like facial expressions can enrich the interaction between humans and robots, and add to a robot's ability to convey intention.

Many HRI researchers have used a range of facially expressive robots in their work that offer differing levels of expressivity, facial degrees-of-freedom (F-DOF), and aesthetic appearance. However, because each robot has different hardware, it is challenging to develop transferable software for facial expression synthesis.

Another challenge in the community is that many researchers need to hire animators or Facial Action Coding System (FACS)-trained coders to generate precise, naturalistic facial expressions for their robots or animated faces. This process is expensive in terms of cost and time, especially for synthesizing dynamic expressions. Thus, there is a need in the community for an open-source software system that enables low-cost, naturalistic facial expression synthesis, regardless of the number of F-DOFs in a robot.

Our work offers the HRI community an approach from which they can easily "plug and play" performance-driven[1] and other synthesis techniques on their robot's face, irrespective of its F-DOFs. While other researchers have explored synthesis techniques for facially expressive robots (c.f. [1]), their software may not be readily applicable to other robots since they are platform-dependent, and often incorporate closed-source components.

HRI'15 Extended Abstracts, Mar 02-05 2015, Portland, OR, USA
ACM 978-1-4503-3318-4/15/03.
http://dx.doi.org/10.1145/2701973.2702053.

2. SYNTHESIS METHOD

While in the future we will explore other synthesis techniques, to begin we implemented an end-to-end ROS module that enables performance-driven synthesis on a robot (see Fig. 1). Nodes in the module include:

S, a sensor to capture a performer's face

P, a point streamer which extracts some facial points

F, a feature processor which extracts some features from the streamed facial points

T, a translator to convert the extracted features to either servo motor commands on a physical platform or control points on a simulated face

C, a control interface to control either the animation of a virtual face or motors on a robot

There are multiple ways to implement this concept. In our implementation, S was a webcam that publishes image data to the `/image` topic. P was an open-source, constrained local model (CLM) based face tracker [7], which we ported to ROS (`ros_clm`). `ros_clm` subscribes to the `/image` topic, tracks a mesh with 68 facial points over time, and publishes 2D coordinates of the points to the `/points` topic.

F subscribes to `/points`, measures the distance of each facial point to the nose tip, saving 68 distances in a vector. It then publishes the computed features to the `/features` topic. T subscribes to `/features`, and maps the features to corresponding motor (servo) values on a robotic face.

For each of the servo motors, we found a group of one or more CLM facial points whose movement significantly affected the motor in question. The movements of the points within each group were averaged and we converted these values to servo motor commands using the approach proposed

[1]This technique involves using live or recorded motion from an actor, and mapping the motion to synthetic face.

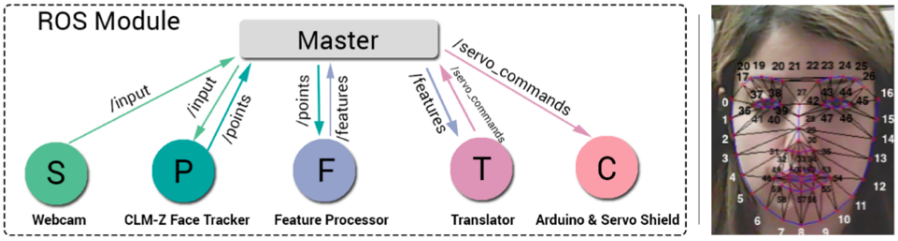

Figure 1: Left: Overview of our synthesis technique as implemented in ROS. Right: An example of *P*.

by Moussa et al. [4]. *T* publishes these values to the topic `/servo_commands`. *C* subscribes to the topic `/servo_commands` and sends the movement commands to the robot.

To validate our approach, we ran two experiments, with the goal of ensuring the synthesized faces were comprehensible to people. The first was conducted in simulation, and the second on our 16 F-DOF bespoke robot head.

2.1 Evaluation 1: Simulation

We describe these details in our other work [3], but will summarize briefly here. We extracted source videos of pain, anger, and disgust from the UNBC-McMaster Pain Archive [2] and MMI database [5]. We selected these expressions for two reasons. First, we are building robotic patient simulator systems, and are thus interested in naturalistic pain synthesis [3]. Using expressions commonly conflated with pain (anger and disgust) enable us to perform a thorough evaluation. Second, we were replicating a pain synthesis study by Riva et al. [6, 3].

We mapped nine source videos (three per expression) to three virtual faces using the Steam Source SDK, resulting in 27 stimuli videos. We conducted an online study (n=50). Participants watched the stimuli videos and labeled the expressions. They were able to identify expressions synthesized by our method; overall accuracy: Pain: 67.33%, Anger: 64.89%, and Disgust: 29.56%[2].

Riva et al. [6] manually synthesized painful facial expressions on a virtual avatar with the aid of an animator, and found 60.4% as the overall pain labeling accuracy [6]. We found our synthesis method achieved arithmetically higher labeling accuracies for pain, which was encouraging.

2.2 Evaluation 2: 16-F-DOF Robot

As part of our patient simulation research, we are building a 16 F-DOF robot. We used this as the platform in our second evaluation. Since the skin of our robot is still under development, we did not run as full a perceptual study as we performed in simulation. However, as we were testing how the control points on the robot's head moved in a side-by-side comparison to a person's face, we do not believe this was especially problematic for this evaluation.

A performer sat in front of a webcam, and performed 10 face-section expressions two times each, yielding 20 videos. The expressions were: *neutral*, *raise eyebrows*, *frown*, *look right*, *look left*, *look up*, *look down*, *raise cheeks*, *open mouth*, *smile*. Our robot mimicked each expression in real time.

In the experiment, we showed participants side-by-side videos of the person and robot making each face-section expression. We asked participants to rate the similarity and the synchrony of the performer's expressions with the robot's using a 5-point Discrete Visual Analogue Scale.

Participants (n=12) were all university students, mean age 22 yrs. The overall average score for similarity and synchrony between the robot and the performer expressions were 3.85 (s.d.=1.28), and 4.30 (s.d.=1.01) respectively. The relatively high overall scores of similarity and synchrony suggest that our method can accurately map facial expressions of a performer onto a robot in real time.

[2]Low disgust accuracies are not surprising; it is known to be a poorly distinguishable in the literature.

3. DISCUSSION AND FUTURE WORK

We described a generalized method for facial expression synthesis on robots, and its implementation in an open-source, end-to-end ROS module. In one validation implementation, CLM-based performance-driven synthesis, our method can accurately map both live and recorded performers to a 16-F-DOF robotic face.

The module is easily extensible to multiple platforms, regardless of the sensor, feature processor, translator, or control interface. This will be beneficial to the HRI community and beyond, as it does not require expensive or labor-intensive animators, and is easy to adapt and operating with very little work on the part of the researcher.

4. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1253935.

5. REFERENCES

[1] P. Jaeckel, N. Campbell, and C. Melhuish. Facial behaviour mapping from video footage to a robot head. *Robotics and Autonomous Systems*, 2008.

[2] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews. Painful data: The UNBC-McMaster shoulder pain expression archive database. In *FG 2011*.

[3] M. Moosaei, M. J. Gonzales, and L. D. Riek. Naturalistic pain synthesis for virtual patients. In *IVA 2014*.

[4] M. B. Moussa, Z. Kasap, N. Magnenat-Thalmann, and D. Hanson. MPEG-4 FAP animation applied to humanoid robot head. *Proceeding of Summer School Engage*, 2010.

[5] M. Pantic, M. Valstar, R. Rademaker, and L. Maat. Web-based database for facial expression analysis. In *ICME '05*.

[6] P. Riva, S. Sacchi, L. Montali, and A. Frigerio. Gender effects in pain detection: speed and accuracy in decoding female and male pain expressions. *EUR J PAIN*, 2011.

[7] J. M. Saragih, S. Lucey, and J. F. Cohn. Face alignment through subspace constrained mean-shifts. In *ICCV '09*. IEEE, 2009.