Interaction and Learning in a Humanoid Robot Magic Performance

Kyle Morris and **John Anderson** and **Meng Cheng Lau** and **Jacky Baltes**
University of Manitoba, Winnipeg, MB R3T2N2, Canada

Abstract

Magicians have been a source of entertainment for many centuries, with the ability to play on human bias, and perception to create an entertaining experience. There has been rapid growth in robotics throughout industrial applications; where primary challenges include improving human-robot interaction, and robotic perception. Despite preliminary work in expressive AI, which aims to use AI for entertainment; there has not been direct application of fully embodied autonomous agents (vision, speech, learning, planning) to entertainment domains. This paper describes preliminary work towards the use of magic tricks as a method for developing fully-embodied autonomous agents. A card trick is developed requiring vision, communication, interaction, and learning capabilities all of which are coordinated using our script representation. Our work is evaluated quantitatively through experimentation, and qualitatively through aquiring 2nd place at the 2016 IROS Humanoid Application Challenge. A video of the live performance can be found at https://youtu.be/OMpcmcPWAVM.

Introduction

Humans have long enjoyed the clever trickery that comes from a good magic show. Magic tricks embody the primary features desired for an intelligent agent. These include **reactivity**: the ability to quickly perceive and respond to changes in the environment; **proactivity**: being goal-driven and acting towards reaching some desired goal; and **social ability**: the ability to communicate with others to further reach their goal (Wooldridge 2009).

Non-deterministic and dynamic environments pose challenges in developing robust autonomous agents that possess these features. This difficulty lies in balancing the proactive and reactive behaviour (Wooldridge 2009). An agent that is purely reactive may fail to reach a desired goal, whereas a purely proactive (goal driven) agent may not spend enough time acting to reach a goal (Wooldridge 2009).

During a live performance, reactivity is desired to provide authentic response time for each event in the script. Proactiveness involves seeking an end-performance goal that logically entails from the events in the script. The script is central to both reactivity and proactiveness. Lastly, social ability is required to leverage off the audience and guide a performance to cater towards their demographic and play off of their bias. For example, non-explicit humorous remarks are prioritized for an audience containing youth. Our work presents an autonomous agent that performs a magic card trick. We created motion, speech, and vision components on top of our custom DARwIn OP2 framework. These components utilize PocketSphinx for speech recognition, and OpenCV2 for playing card classification. The use of a finite state machine gives structure to the performance and allows the agent to seek an end-performance goal that accounts for potential problems that may arise during the show. Lastly, an easily adjustable design of events allows for a unique performance and user experience.

Related Work

Live performance takes many forms. Humanoid robotics competitions have explored the development of robust, versatile agents that perform multiple distinct sporting events autonomously (Baltes et al. 2016). Furthermore, teams of robots are used to research how cooperation techniques are used in reaching a desired goal (Ashar et al. 2015). Such competitions have grown in popularity and have evolved to use more entertaining events that remain as useful benchmarks (Gerndt et al. 2015), but do not yet cater easily to a non-research audience.

Expressive AI has explored artificial intelligence for pure entertainment purposes in domains that include games (eag 2003) and music (De Mántaras and Arcos 2002); but lacks a robotics implementation. In the domain of Robotics, work has been done on incorporating entertainment (Kuroki 2001) with further specialization into card magic (Koretake, Kaneko, and Higashimori 2015). This work however puts focus on card manipulation, and mechanical aspects rather than timing and interaction. There has been growing discussion of the need for timing and human-robot interaction for effective live performance (Nuñez et al. 2014; Tamura, Yano, and Osumi 2014); but this discussion has been purely theoretical. Our work outlines a new application of robot entertainment for live magic that incorporates computer vision, machine learning, speech recognition and motion in order to deliver an authentic and robust performance.

Employing template-matching for playing card recognition has demonstrated higher overall classification accuracy, but only in settings where the card is viewed from a fixed distance and angle (Brinks and White 2007). Similarly, this approach had significant latency (6 seconds) using a client-server architecture and has not yet been tested on a localized model (Brinks and White 2007). Work from (Zheng and Green 2007) demonstrated higher rank classification accuracy along with robustness to card rotation and scale, however there is no evaluation of the overall classification accuracy. Furthermore we achieved higher accuracy on Jack, Queen, and King cards, along with higher suit accuracy. Other approaches such as (Martins, Reis, and Teófilo 2011) achieved higher rank classification; but share similar challenges in suit classification. Despite marginally lower performance on rank classification, our system demonstrates significant overall classification accuracy while being robust to card rotation, translation, and scale.

The Magic Trick

The trick is based on the classic straight-man act, in which a stern robot assistant contrasts with a charismatic but condescending human magician. A DARwIn-OP2 robot is asked to select and observe 3 cards from a deck. Vocal cues from the human magician provoke responses from the robot. Throughout the performance the robot grows impatient with the magicians' rude gestures and treatment, and takes over the magic performance by knocking the deck out of the magicians' hand. After the robot acquires the deck, the robot explains the simplicity of the magic trick, and reveals the 3 cards that were originally chosen, from the face-down deck.

Figure 1: The live performance at IROS 2016. The robot is about to reveal the cards.

Problem Representation

We represent a performance as a collection of ordered phases. A phase is some discrete set of events that must take place together within a limited time. For example one phase may involve multiple listen-response events where an agent uses speech recognition and speech synthesis to follow dialog with a human magician. Another phase may rely on both motion gestures to hold a deck of cards, and computer vision to recognize playing cards.

Grouping events into phases allows for a graceful recovery from potential interrupts in the performance. If, for example, a dialog-only phase is taking place, and noise interference occurs, the agent may transfer to a backup phase

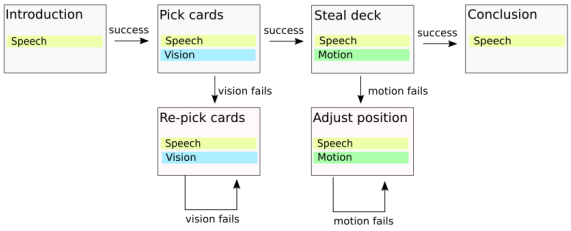

Figure 2: Phases of the performance

which involves asking where the noise is coming from. During a card recognition phase that uses only the vision and motion components, it would not make sense for the agent to stop reading cards, or freeze up; because of the noise. It would make sense to have a backup phase in case the lighting is poor, in which the agent may ask for better lighting. The use of a state machine guides the performance by transitioning through pre-designed phases which together form a coherent story.

Implementation

Speech Recognition and Synthesis

Voice audio was recorded using a NESSIE Adaptive USB Condenser Microphone at 16kHz. Incoming audio is processed using PocketSphinx in order to generate a hypothesis string. This hypothesis string is checked against a custom language dictionary containing 89 keywords from the magic show script. If selected keywords are found in said string, this will trigger a response from the robot. Each dialog event may be customized to require multiple distinct keywords.

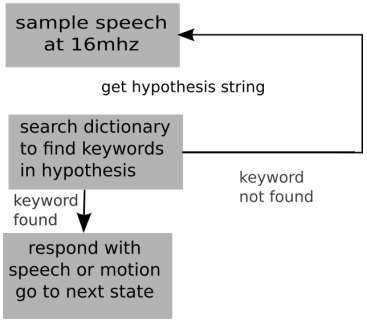

Figure 3: Control flow of speech processing

Vision

Input images are captured using the built-in DARwIn-OP2 Logitech camera and passed to a custom vision module. The vision module was built with C++ and OpenCV2. The input image is first preprocessed by gray scaling, applying blur, and then applying a binary threshold. Contours are then extracted from the image and organized into a hierarchical tree

and compressed with OpenCV's simple chain approximation to gather only end-points of the contours. Polygon approximation is used on the contour to gather estimated corner points for a playing card. In order to eliminate false detection, the points are checked to be rectangular(based on the ratio between them). An affine transformation is used on the card ROI. Due to symmetry of playing cards, the bottom left corner is checked for a card symbol. If this symbol is missing, the card is assumed to be mirrored, and will be reflected to the correct orientation.

Card Classification Card suit (Diamonds, Hearts, Spades, Clubs) and rank (1-10, Jack, King, Queen, Ace) ROI are extracted. These ROI are then either dilated or eroded according to lighting in the environment. The suit and rank ROI are then classified using the K-Nearest Neighbours algorithm (Cover and Hart 1967).

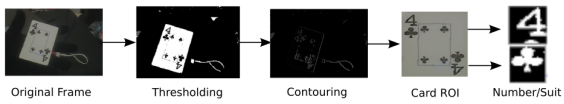

Figure 4: The vision pipeline

Machine Learning

The training process took place using a deck of 52 cards. The initial training set contained $5_{images} \times 4_{suits} \times 13_{cards} = 260$ samples collected using the robots built-in camera. Each sample is stored as a 30x30 gray-scale image in csv format as a 1×900 matrix of pixel brightness values [0-255]. The K-Nearest Neighbours algorithm (Cover and Hart 1967) is used to classify each suit and rank. An iterative training process is

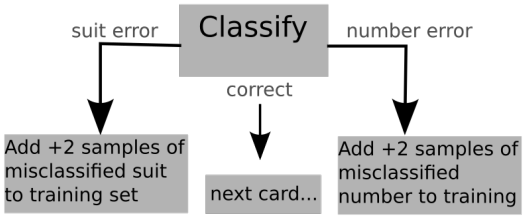

Figure 5: The training process

used. Initially each card within the full deck is shown in front of the robot. If the card is correctly classified, it is placed in a success pile. Misclassification may take place on either the card rank or suit. In either case, the misclassification is recorded and 2 positive samples of this rank or suit are added to the training set. The card will then be placed in a fail pile. For example if a Two of Hearts is misclassified as a Two of Diamonds, we will add 2 positive samples of the Hearts suit to the training set. The next iteration will begin using cards from the fail pile. This iterative process terminates when the fail pile is empty.

Evaluation

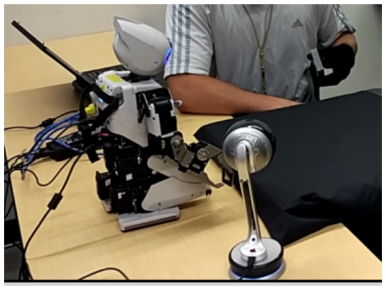

Figure 6: The dynamic evaluation setup.

Our iterative training process was used, yielding the final training set. The test set was then created by randomly shuffling the deck and placing each card in front of the robot. This process was repeated 5 times to create a total of $5_{samples} \times 13_{ranks} = 65$ test samples for each suit, and $5_{samples} \times 4_{suits} = 20$ test samples for each rank. Evaluation was first completed in a dynamic setting. This included exposure to daylight, and randomization from a human holding the card in front of the robot. A second controlled evaluation consisted of static lighting, and a fixed placement of each card on a black surface.

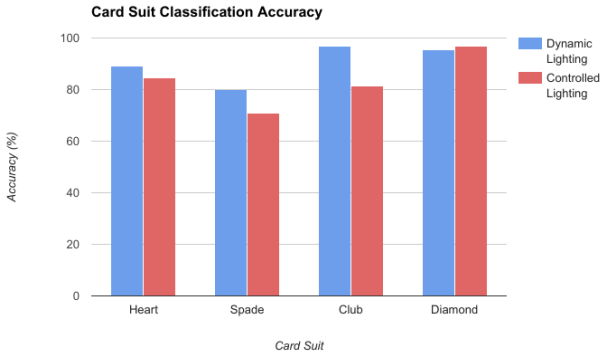

Figure 8: Classification results for card suits. Taken from 60 samples of each card suit.

A rank classification accuracy of 89.23% across the 13 card ranks was achieved using the dynamic setting. This surpassed the controlled setting which achieved 83.46% accuracy. Similarly the dynamic setting achieved a higher classification accuracy (90.38%) than the controlled setting (83.46%) on card suits. It is interesting to note the difference in spread between the two evaluations. The controlled setting has a higher standard deviation (10.76% for card rank, 15.99% for card suit) than the dynamic setting (4.07% for card rank, 11.15% for card suit). We believe this is due to our system being trained in a more dynamic setting.

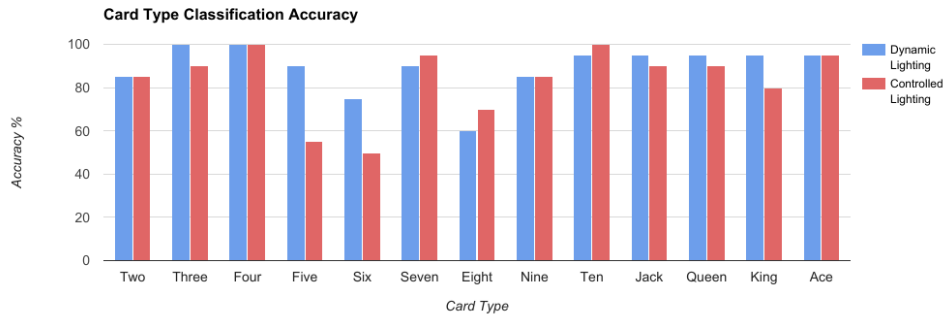

Figure 7: Classification results for card ranks. Taken from 20 samples of each card rank.

Conclusions and Future Work

This work explored the use of live entertainment in agent-based research. Specifically live magic performance was chosen as an avenue for developing a fully-embodied autonomous agent. Our card trick incorporates on-board vision, communication, interaction, and learning capabilities that allow for robust performance. This work may be greatly enhanced with improvements to the vision and machine learning components. Overall classification accuracy is dependent on both rank and suit accuracy. Our method demonstrated robustness to card rotation, translation and scale; but fell short in overall accuracy. We share similar challenges to other aforementioned vision techniques (Brinks and White 2007; Zheng and Green 2007; Martins, Reis, and Teófilo 2011), and believe improvements to image resolution would combat these challenges. Similarly, we see the use of colour recognition as a simple and promising approach to improve suit classification accuracy (Martins, Reis, and Teófilo 2011). Such improvements are challenging to acquire under time and space constraints imposed by on-board hardware. Lastly, we are interested in generalizing our work into a framework for building agents capable of live performance. We believe this framework would provide easier entry, and thus encourage agent-based research using live entertainment.

References

Ashar, J.; Ashmore, J.; Hall, B.; Harris, S.; Hengst, B.; Liu, R.; Mei (Jacky), Z.; Pagnucco, M.; Roy, R.; Sammut, C.; Sushkov, O.; Teh, B.; and Tsekouras, L. 2015. *RoboCup SPL 2014 Champion Team Paper*. Cham: Springer International Publishing. 70–81.

Baltes, J.; Tu, K.-Y.; Sadeghnejad, S.; and Anderson, J. 2016. Hurocup: competition for multi-event humanoid robot athletes. *The Knowledge Engineering Review* 1–14.

Brinks, D., and White, H. 2007. Texas hold 'em hand recognition and analysis.

Cover, T., and Hart, P. 1967. Nearest neighbor pattern classification. *IEEE transactions on information theory* 13(1):21–27.

De Mántaras, R. L., and Arcos, J. L. 2002. Ai and music: From composition to expressive performance. *AI Magazine* 23:43–58.

2003. Expressive ai: Games and artificial intelligence. In *DiGRA '03 - Proceedings of the 2003 DiGRA International Conference: Level Up*.

Gerndt, R.; Seifert, D.; Baltes, J. H.; Sadeghnejad, S.; and Behnke, S. 2015. Humanoid robots in soccer: Robots versus humans in robocup 2050. *IEEE Robotics & Automation Magazine* 22(3):147–154.

Koretake, R.; Kaneko, M.; and Higashimori, M. 2015. The robot that can achieve card magic. *ROBOMECH Journal* 2(1):5.

Kuroki, Y. 2001. A small biped entertainment robot. In *MHS2001. Proceedings of 2001 International Symposium on Micromechatronics and Human Science (Cat. No.01TH8583)*, 3–4.

Martins, P.; Reis, L. P.; and Teófilo, L. 2011. Poker vision: playing cards and chips identification based on image processing. In *Iberian Conference on Pattern Recognition and Image Analysis*, 436–443. Springer.

Nuñez, D.; Tempest, M.; Viola, E.; and Breazeal, C. 2014. An initial discussion of timing considerations raised during development of a magician-robot interaction. In *Proc. ACM/IEEE Workshop on Timing in Human-Robot Interaction HRI*.

Tamura, Y.; Yano, S.; and Osumi, H. 2014. Modeling of human attention based on analysis of magic. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-robot Interaction*, HRI '14, 302–303. New York, NY, USA: ACM.

Wooldridge, M. 2009. *An introduction to multiagent systems*. John Wiley & Sons.

Zheng, C., and Green, R. 2007. Playing card recognition using rotational invariant template matching. In *Proc. of Image and Vision Computing New Zealand*, 276–281.