HeroMirror Interactive: A Gesture Controlled Augmented Reality Gaming Experience

Tamás Matuszka
Department of Research & Development
INDE R&D
tamas@indestry.com

Ferenc Czuczor
Department of Software Development
INDE R&D
feri@indestry.com

Zoltán Sóstai
Department of Content Development
INDE R&D
zoltan@indestry.com

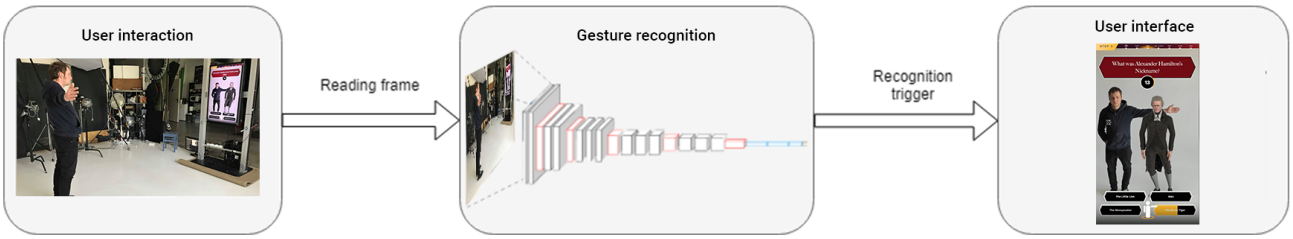

Figure 1: HeroMirror Interactive system overview. The player is enabled to select an answer of the quiz with a gesture.

ABSTRACT

Appropriately chosen user interfaces are essential parts of immersive augmented reality experiences. Regular user interfaces cannot be efficiently used for interactive, real-time augmented reality applications. In this study, a gesture controlled educational gaming experience is described where gesture recognition relies on deep learning methods. Our implementation is able to replace a depth-camera based gesture recognition system using conventional camera while ensuring the same level of recognition accuracy.

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI)**; • **Computing methodologies** → **Computer vision**; **Object detection**; **Neural networks**;

KEYWORDS

augmented reality, deep learning, human experience, computer vision

ACM Reference Format:
Tamás Matuszka, Ferenc Czuczor, and Zoltán Sóstai. 2019. HeroMirror Interactive: A Gesture Controlled Augmented Reality Gaming Experience. In *Proceedings of SIGGRAPH '19 Posters.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3306214.3338554

SIGGRAPH '19 Posters, July 28 - August 01, 2019, Los Angeles, CA, USA

ACM ISBN 978-1-4503-6314-3/19/07.
https://doi.org/10.1145/3306214.3338554

1 INTRODUCTION

Augmented reality (AR) applications are rapidly gaining popularity and have matured to the phase when their usage such as face filtering (Snapchat[1]) or gaming (Pokemon Go[2]) is not considered to be futuristic. However, meaningful interaction with these applications cannot be provided by means of regular user interfaces due to the nature of AR experiences. Since interactivity is part of the definition of AR [Azuma 1997], an efficient and easy-to-use interaction method has to be provided. In the last few years, several solutions have been developed in order to overcome this issue [Wang et al. 2015], [Chen et al. 2015], [Da Gama et al. 2016]. The majority of these solutions rely on additional hardware (e.g. depth camera, Kinect sensor) that increases the complexity and costs of these systems.

In this paper, a deep learning-based gesture recognition method is described that allows the detection of predefined gestures using a regular camera. The captured frames are processed by a state-of-the-art convolutional neural network (CNN) trained for object detection. We have utilized the aspect ratio deformation of the bounding box of the player to recognize the predefined gestures. Due to this method, the need for computationally expensive pose estimation or recurrent neural network usage can be avoided and the problem can be defined as activity recognition with a CNN. Furthermore, since a pre-trained object detector is suitable for the task, we could avoid transfer learning too. In this way, the saved GPU capacity can be used for rendering high-quality animation in augmented reality.

As the validation of the feasibility of our method, we have implemented an augmented reality kiosk, allowing players to have a device-free, interactive and educational experience with a virtual

[1] https://www.snapchat.com/
[2] https://pokemongolive.com/en/

character. Controlling the experience via gesture, players enter a virtual quiz that can be guided by their selected "hero".

2 TECHNICAL APPROACH

2.1 Gesture recognition algorithm

The core idea of our gesture recognition algorithm is that computationally relatively inexpensive object detection performed by a CNN can be used for recognizing predefined gestures. In this way, we do not have to rely on more complex pose estimation or instance segmentation algorithms. The proposed method can be divided into two parts. The first part applies object detection, namely person detection on each frame. Using the initial bounding box of a user obtained by the calibration process, the deformation of the player bounding box in the camera frame allows us to determine whether a predefined gesture has been performed (left arm upwards, right arm upwards, left arm sideways, right arm sideways have been chosen as gestures). If the change of aspect ratio, width, and height of the bounding box with respect to the initial parameters fall between empirically experimented boundaries, the gesture can be determined for each frame (e.g. a raised hand increases the height, a sideways hand increases the width).

The second part of the algorithm is a sliding window method that collects the classified gestures for each frame. The sliding window method has two parameters, the s size of the window and the ϵ threshold. If the number of the same gestures within the window exceeds the threshold, then the gesture recognition can be triggered. The recognition speed and accuracy can be controlled by the parameters of the algorithm. The bigger is s and ϵ, the slower is the gesture detection algorithm. However, the accuracy can be increased with bigger parameter values.

2.2 Visualization and content creation

As the evidence of the applicability of the proposed gesture recognition method, a human gesture-driven "Who Wants to be a Millionaire" style AR quiz game has been implemented. The player of the game is allowed to choose the correct answer by performing one of the four predefined hand gestures. A virtual "hero" character moderates the game, as they are presented on screen standing next to the player using AR. At the end of the quiz, the "hero" character presents the final score and the reward: two photos. Finally, visitors can choose a picture and share it on social media, email it to a personal email address or get a printout.

Alexander Hamilton, one of the Founding Fathers of the United States has been selected as the hero in our reference implementation. The model was created with Maya, the animations have been developed with Rokoko motion capture suit and Unity.

2.3 Results and Evaluation

The gesture classification part of the algorithm has been implemented in Python language using a RetinaNet [Lin et al. 2017] Keras[3] implementation with ResNet [He et al. 2016] backbone and TensorFlow[4] was used as the deep learning backend. The boundaries for classifying the frames to specific gesture have been determined by empirical experiments. Then, the classification result is sent to Unity via socket communication where the sliding window-based method and the visualization are implemented using C# language. We have set $s = 9$ and $\epsilon = 0.85$ in order to ensure fast and accurate gesture recognition. The sliding window has been implemented with a double-ended queue data structure. The current implementation of the proposed system runs in near real-time (<1s recognition time is feasible) on a PC with an i7 CPU, 16 GB RAM, NVIDIA GeForce GTX 1070 GPU, and a Logitech BRIO webcam.

We have developed a Kinect reference implementation and used it to compare with our results in terms of gesture recognition accuracy. The recognition was used in the above-mentioned gaming context, and the experiments have been conducted with three subjects. Both Kinect-based solution and our method were able to accurately recognize the gestures of subjects (one game has included four questions).

3 CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a monocular camera based gesture recognition method that can ensure similar level of recognition accuracy as depth-camera based solutions. In addition, an augmented reality gaming experience has been built on the top of our proposed method. Our end-to-end solution innovatively combines augmented reality and deep learning, applying the latest results of these research areas in order to provide an immersive, individually tailored, and interactive AR experience. In the future, we are aiming to implement more recognizable gestures and conduct more in-depth evaluation and comparison.

ACKNOWLEDGMENTS

The authors would like to thank Márton Helényi for animation development, Norbert Kovács for user interaction recommendations, and Dániel Siket for reference image creation. The authors would also like to thank the anonymous referees for their valuable comments and helpful suggestions.

REFERENCES

Ronald T Azuma. 1997. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments* 6, 4 (1997), 355–385.

Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. 2015. Improving human action recognition using fusion of depth camera and inertial sensors. *IEEE Transactions on Human-Machine Systems* 45, 1 (2015), 51–61.

Alana Elza Fontes Da Gama, Thiago Menezes Chaves, Lucas Silva Figueiredo, Adriana Baltar, Ma Meng, Nassir Navab, Veronica Teichrieb, and Pascal Fallavollita. 2016. MirrARbilitation: A clinically-related gesture recognition interactive tool for an AR rehabilitation system. *Computer methods and programs in biomedicine* 135 (2016), 105–114.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

Chong Wang, Zhong Liu, and Shing-Chow Chan. 2015. Superpixel-based hand gesture recognition with kinect depth camera. *IEEE transactions on multimedia* 17, 1 (2015), 29–39.

[3] https://keras.io/

[4] https://www.tensorflow.org/