International Journal of Advanced Culture Technology Vol.9 No.4 295-301 (2021)
DOI https://doi.org/10.17703/IJACT.2021.9.4.295

IJACT 21-12-35

Augmented Reality Service Based on Object Pose Prediction Using PnP Algorithm

[1]In-Seon Kim, [2]Tae-Won Jung, [3]Kye-Dong Jung

[1]Department of Smart System, Kwangwoon University,
20 Kwangwoon-ro, Nowon-gu, Seoul 01897, Korea
[2]Department of Immersive Content Convergence, Kwangwoon University,
20 Kwangwoon-ro, Nowon-gu, Seoul 01897, Korea
[3]Ingenium College of Liberal Arts, Kwangwoon University,
20 Kwangwoon-ro, Nowon-gu, Seoul 01897, Korea
{kisidid, onom, gdchung}@kw.ac.kr

Abstract

Digital media technology is gradually developing with the development of convergence quaternary industrial technology and mobile devices. The combination of deep learning and augmented reality can provide more convenient and lively services through the interaction of 3D virtual images with the real world. We combine deep learning-based pose prediction with augmented reality technology. We predict the eight vertices of the bounding box of the object in the image. Using the predicted eight vertices(x,y), eight vertices(x,y,z) of 3D mesh, and the intrinsic parameter of the smartphone camera, we compute the external parameters of the camera through the PnP algorithm. We calculate the distance to the object and the degree of rotation of the object using the external parameter and apply to AR content. Our method provides services in a web environment, making it highly accessible to users and easy to maintain the system. As we provide augmented reality services using consumers' smartphone cameras, we can apply them to various business fields.

Keywords: *Augmented Reality, Deep Learning, Mobile Web Service, Object Detection, Pose Prediction*

1. INTRODUCTION

The development of convergence quaternary industrial technology and mobile devices causes research on related technologies and the development of services using them. Augmented reality technology is one of the digital media technologies. The interaction between the real world and the 3D virtual image can maximize the experience of visual and auditory information[1]. With the commercialization of 5G, augmented reality-related technologies are expected to be used in more fields[2]. The combination of image recognition and augmented reality using deep learning delivers information more conveniently and vividly to users. The combination of CNN-based object recognition and augmented reality is already being used in various fields[3-5]. Various standard integrated platforms for the reality of expansion in mobile devices are being studied. In the process of implementing the app-based expansion reality, there were inconveniences due to differences in the development environment, and as a result, the need for a standard integrated platform to implement the expansion reality in the web environment emerged. W3C Group unveiled the WebXR Device API in 2018 and

Manuscript received: October 26, 2021 / revised: November 10, 2021 / accepted: December 7, 2021
Corresponding Author: gdchung@kw.ac.kr
Tel: +82-2-940-5288
Professor, Ingenium College of liberal arts, Kwangwoon University, Seoul, Korea

is still actively developing it until recently. The WebXR Device API supports both virtual reality and augmented reality on the web and supports various related functions. It manages the selection of the output device and renders a 3D scene at an appropriate frame speed to the selected device[6].

Our method combines pose prediction using deep learning and augmented reality technology. The system acquires an image from a screen viewed by the user. It transmits the acquired image to the server to predict the location and rotation of the object. The position of the predicted object determines the position of the AR. The predicted rotation rotates the AR model. It provides more vivid AR services to users by applying location and rotation in this way.

Section 2 describes related research. Section 3 describes the flow chart and overall overview of the proposed system. Section 4 describes an example of combining object pose prediction and augmented reality by applying the proposed system. Section 5 describes the system summary and future research tasks.

2. RELATED RESEARCH

2.1 Object Pose Prediction

We use object detection using deep learning in many fields. Many services already utilize object recognition and augmented reality using CNN in various fields[3-5]. The 2D boundary box predicted by object detection provides only the maximum and minimum values of X and Y. AR can also be provided with the information. However, by using a 6D pose including a rotation value of an object, the AR model can be rotated according to the degree of rotation of the object, thereby providing a more realistic augmented reality service[7]. With the advent of commercial depth cameras, many RGB-D object pose estimation methods have emerged. However, most mobile devices have not yet built-in depth cameras. There are many restrictions on mobile models to be applied to general-purpose services. We use a neural network capable of predicting 6D poses of objects using a single CNN-based RGB image [8]. It is based on YOLO and predicts the 2D position in which the edge of the object's 3D boundary box is projected from the RGB image. 6D poses are calculated through the PnP algorithm using 2D projection, camera intrinsic parameters, and 3D mesh[9].

2.2 WebXR Device API

WebXR Device API is an API developed to provide extended reality(XR) services in a web browser[6, 10, 11]. In general, accessibility is poor because in order to use extended reality on mobile devices, applications suitable for the operating system must be downloaded in advance. WebXR provides services through a web browser, so there is no need to download in advance. With the development of communication technology, AR contents with large capacity can also be serviced in real time. When providing an AR service using the WebXR Device API, camera rights are acquired and used in a web browser. WebXR Device API accesses immersive-session when providing AR services, and in this case, images for object detection cannot be obtained because camera rights are used independently. Therefore, before accessing the immersion session, we acquire an image for object detection and transmit it to the server. When detection is completed, AR services are provided by accessing the immersive-session.

3. THE PROPOSED SYSTEM

3.1 The Flow Chart of the Proposed System

Figure 1 shows the flow chart of our method. The user accesses the web page providing the service. The image acquired by the user's camera is transmitted to the server using the websocket. The server preprocess

the received image to enter the prediction layer. When the preprocessed image passes through the prediction layer, the 2D position of the corner of the 3D boundary box expected in the image is predicted. The PnP algorithm is applied using the camera intrinsic parameters of the smartphone entered in advance, the 3D mesh model of the object, and the position of the predicted boundary box edge. PnP provides 2D coordinates projected for the rotation value, movement value, and 3D boundary box edge of the object at camera coordinates. The position of the AR is calculated using the coordinates and movement values of the projected boundary box of the object in the client. The rotation of the AR is calculated using the rotation value of the object. AR service whose position and rotation are determined is provided to the user.

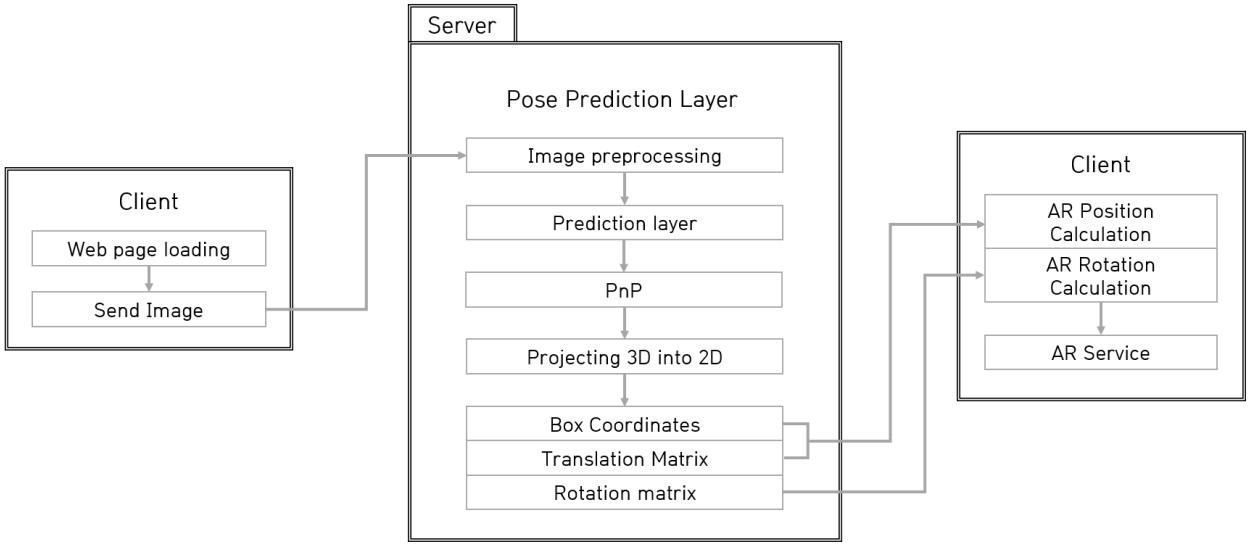

Figure 1. The flow chart of the proposed system

3.2 System Module Design

3.2.1 Client Module Design

Figure 2 shows a module operated by a client when a user accesses a web page providing a service. It first checks whether the user's camera can be used. If the camera is available, it opens the camera after setting the camera to be used on the service page. It sets Three.js to render AR content and prepares AR content to be shown to the user in advance. When both the camera and AR content are prepared, it communicates with the server using a web socket. When the connection with the server is successful, it acquires an image using a user camera and transmits the image to the server.

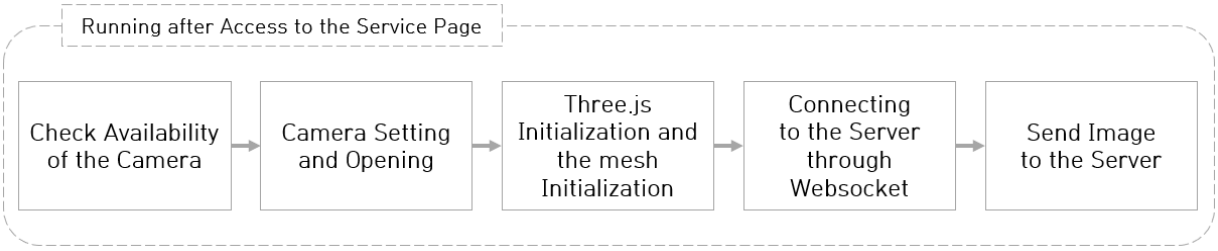

Figure 2. Running after access to the service page

Figure 3 shows the module operated by the client after pose prediction of the object is completed in the image transmitted to the server in Figure 2. It receives information about the predicted pose of the object from the server. Here, using the predicted coordinates of the box and the moving value of the camera, it calculates the location to provide the AR service, anchor. Among the received information, using the rotation value of the camera, it rotates the AR content. It provides AR content for which movement and rotation are determined to the user.

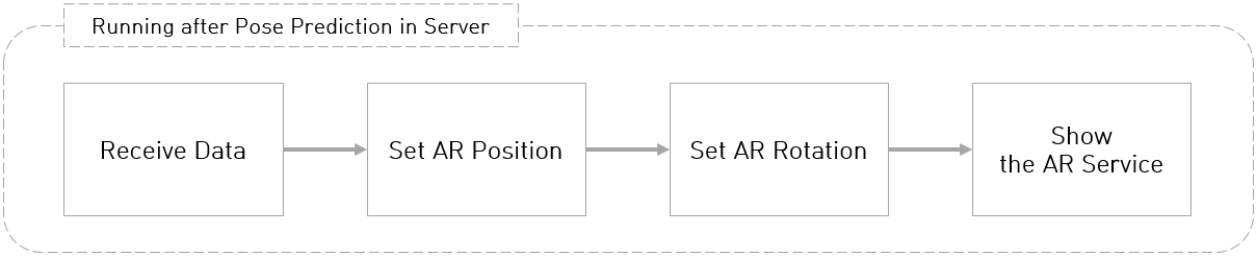

Figure 3. Running after pose prediction in server

3.2.2 Server Module Design

Figure 4 shows a module that operates after the server is executed. It first reads the file constituting the prediction layer, the definition of the object to be detected, and the configuration file containing information from the camera. After reading the 3D mesh information consisting of coordinates in virtual space, it calculates the maximum and minimum values of each x, y, and z to obtain the coordinates of the corners of the truth box of the object. It calculates the camera matrix from the camera information. Finally, it preloads the prediction layer so that prediction can proceed as soon as the image is input.

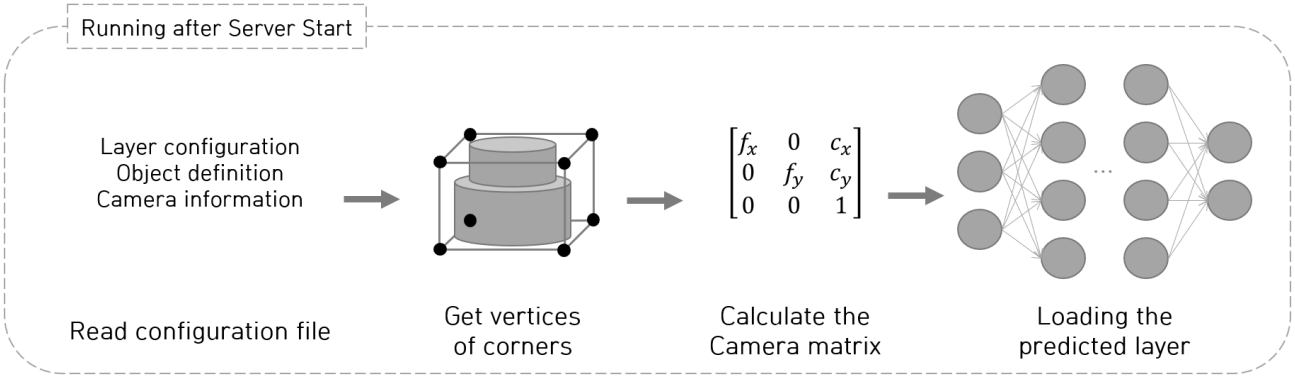

Figure 4. Running after server start

Figure 5 shows a module that operates after receiving an image from a client. It preprocesses the received image before it passes through the prediction layer. When the preprocessed image passes through the prediction layer, it predicts the x and y values of the eight vertices and center points of the rectangular parallelepiped in the image. It performs a PnP algorithm operation using the previously acquired camera matrix, verities of seniors in virtual space, and eight predicted vertices. It acquires verities that project the verities of the seniors in 3D space onto the image using the external parameters of the camera acquired by the PnP algorithm. It transmits the projected vertices, translation values(T_x, T_y, T_z) and rotation values(R_x, R_y, R_z) calculated from the camera's external parameters to the client.

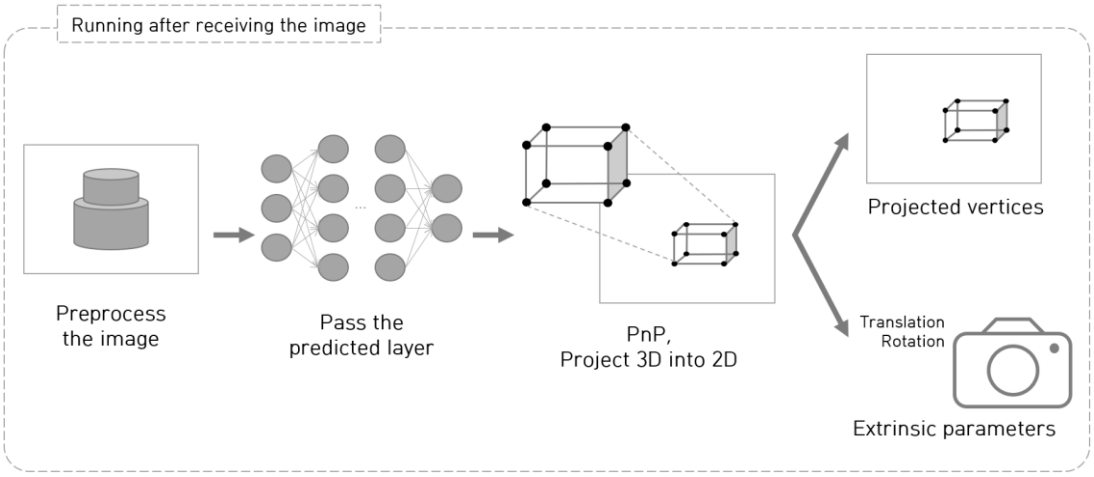

Figure 5. Running after receiving the image

4. SYSTEM APPLICATION

Figure 6 shows the application of our method. We use Samsung Galaxy Note20 5G (SM-N981N) as a client. We use Android 11 as an operating system. We use Chrome 93.0.4577.62 version as a web browser. We use computers with Windows 10, NVIDIA GeForce GTX 1650, and 16GB of RAM as servers. To provide a service, we first receive an image and run a server to predict. The server converts the pre-input camera intrinsic parameters into a matrix, loads the prediction layer, and prepares a connection through the web socket (in(a)). When the user accesses the service page after the server is executed, an image is acquired using a camera. The acquired image is transmitted to the server through the web socket (in(b)). The image received from the server is preprocessed and then predicted. As a result of the prediction, it provides the verities of the box, the camera translation value, and the rotation value, and transmits these values to the client (in(c)). Using the values received from the client, the anchor and rotation of the AR are calculated and shown to the user (in(d)).

Figure 6. Application of the proposed system

5. CONCLUSION

We propose an augmented reality service system using pose prediction of objects and WebXR Device API in a mobile web environment. Since the proposed system takes place in a mobile web environment, it is easy to maintain and repair the system. Since the deep learning prediction operation is performed on the server, the burden on the client is low. Also, we only use images made up of RGB. Since a sensor for acquiring depth information such as ToF is not required, the influence of the smartphone model is small. However, it is not possible to obtain camera intrinsic parameters for using the PnP algorithm on the web. Therefore, the service provider needs to input camera-specific variables according to the model in advance. Since the WebXR Device API used in the proposed system exclusively uses camera rights, there is a problem that object detection and augmented reality services cannot be provided simultaneously in one session. If it is upgraded to allow extraction of screen frames using camera privileges in the immersive-ar mode in the future, it is possible to detect objects in one session, so more convenient services can be provided.

ACKNOWLEDGE

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-00192, Development of Open Anchor-based AR Cloud for Universal Mixed Reality Service)

REFERENCES

[1] Hyung-chul Kang, "A Study on Smart Tourism Content Using Augmented Reality," *A Journal of Brand Design Association of Korea*, Vol.17, No.3, pp. 165 - 174, Sep 2019. DOI: https://doi.org/10.18852/bdak.2019.17.3.165

[2] Ki-Hwan Ko, “A Study on the Graphic Production Technology for AR Augmented Reality Game,” *The Journal of Korean Institute of Information Technology*, Vol. 16, No. 11, pp. 123 – 132, Nov 2018. DOI: https://doi.org/10.5762/KAIS.2020.21.7.262

[3] Hyun-ju Oh, Ji-sol Jung, So-jung Park and Ki-Yong Lee, “Development of a tour information system for smart phones using CNN,” *Journal of the Korean Institute of Information Scientists and Engineers 2019 Korea Computer Science Competition*, pp. 1642-1644, Jun 2019.

[4] Jong-in Choe, Jung-hyun Lee, Dong-wan Kang and Sang-hyun Seo, "AR based Beverage Information Visualization and Sharing System using Deep Learning," *Journal of Digital Contents Society*, Vol. 21, No. 3, pp. 445-452, Mar 2020. DOI: https://dx.doi.org/10.9728/dcs.2020.21.3.445

[5] Jin-Seon Oh and In-Gook Chun, “Implementation of Smart Shopping Cart using Object Detection Method based on Deep Learning,” *Journal of the Korea Academia-Industrial cooperation Society*, Vol. 21, No. 7, pp. 262-269 July 2020. DOI: https://doi.org/10.5762/KAIS.2020.21.7.262

[6] Jin-Kyu Kang, “OpenXR and WebXR in Virtual Augmented Reality,” *Broadcasting and Media Magazine*, Vol. 26, No. 1, pp. 12-18, Jan 2021.

[7] Chi-Seo Jeong, Jun-Sik Kim, Dong-Kyun Kim, Soon-Chul Kwon and Kye-Dong Jung, "AR Anchor System Using Mobile Based 3D GNN Detection," *International Journal of Internet, Broadcasting and Communication*, Vol.13, No.1, pp. 54-60, Feb 2021. DOI: http://dx.doi.org/10.7236/IJIBC.2021.13.1.54

[8] Bugra Tekin, Sudipta N. Sinha and Pascal Fua, “Real-Time Seamless Single Shot 6D Object Pose Prediction,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 292-301, Dec 2018.

[9] Vincent Lepetit, Francesc Moreno-Noguer and Pascal Fua, "EPnP: An Accurate O(n) Solution to the PnP

problem," *International Journal of Computer Vision*, Vo. 81, No. 2, Article number. 155, Feb 2009. DOI: http://dx.doi.org/10.1007/s11263-008-0152-6

[10] Blair Maclntyre and Trevor F. Smith, "Thoughts on the Future of WebXR and the Immersive Web," *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 338-342, Apr 2019. DOI: https://doi.org/10.1109/ISMAR-Adjunct.2018.00099

[11] Dae-Hyeon Lee, "A Study on the Performance Evaluation of WebXR Device API in Augmented Reality Environment of Mobile Web," *Master Thesis, Graduate School of Smart Convergence, Kwangwoon University, Korea*, 2020.