Real-world AR Pet: An Emotional Supporter and Home Guide

Mengchao WU
44181504-7

Master of Engineering

Supervisor: Prof. Jiro Tanaka
Graduate School of Information, Production and Systems
Waseda University

September 2020

Abstract

With the aggravation of the aging population, a series of survival difficulties faced by the elderly, such as the inevitable loneliness and the increasing number of Alzheimer's patients, have become urgent problems to be solved in today's society.

Some researches show that Human-animal interaction is a wonderful way to bring joy to many animal lovers, and even to those who never knew they enjoyed animals. Exposure to pets has also been associated with improving function and decreasing the behavioral and psychological symptoms associated with dementia (BPSD) among older adults. These phenomenons show that pets can act as emotional supporters of people.

This research intends to improved the current AR pet system and to put forward a Real-world AR pet system to accompany and navigate people, especially the senior people. This system aims to give the user a more immersive interaction and a more interesting navigation experience.

Our system mainly concludes three parts. 3D model construction, Real-time interaction, and AR pet home guide. First, we construct a 3D mesh of the object from a series of overlapping photos from various angles. And then, we create the bone structure of the 3D model to allow the object to be animated freely. Next, we design real-time interaction according to the different real objects. Realize multiple interactions between AR pets and the real environment. Our system also includes AR pet navigation. In this part, the AR pet will lead the user home as a home guide after they click the "Back Home" button.

Several experiments performed to verify our approach is an effective approach.

Keywords: Augmented Reality, AR pet, AR Navigation, Object Detection

Acknowledgements

At the beginning, I am very happy to express my heartfelt thanks and gratitude to my supervisor, Pro.Jiro Tanaka, who is a highly respected and responsible mentor with rigorous academic attitude and innovative ideas. During the two years of studying in Waseda, he gave me a lot of academic help and guidance. He led me to study the field of augmented reality and human-computer interaction. During every stage of preparation of this thesis, he spared no effort to give me a lot of valuable suggestions.

Besides, it is grateful to all my dear labmates. They discussed a lot with me and always give me lots of inspiration. Special thanks to Liu Yuzhao and Liu Yuhan for giving me a lot of 3D modeling advice. When my research has a bottleneck, Xu Shihui always encourages me and gives me confidence. In the evaluation stage, everyone is willing to help me make evaluation and give many valuable suggestions.

Finally, I want to thank my parents. During my two years of master's degree, it was they who provided me with financial and spiritual support. They always support me to do anything unconditionally, and tell me that it doesn't matter when I fail. They gave me the courage to try and experience.

Contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Introduction

Augmented Reality(AR) which is a new approach or technology that makes the computer interface invisible and enhances user interaction with the real world [1] becomes more and more popular recently. AR pet system is a system in which the user can raise an AR pet and do interact with this AR pet. Such a system could bring a feeling of companionship to users. Recently, there are several AR pet systems in the Application Store.

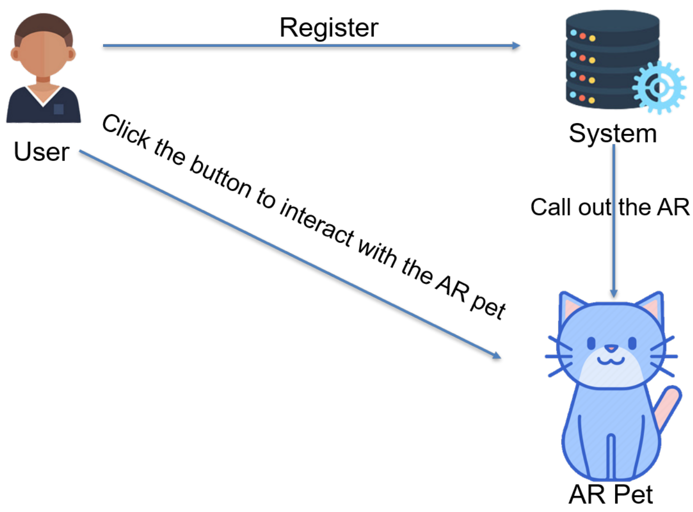


Fig. 1.1 Current AR pet system

Figure 1.1 shows the current AR pet system model. However, these ordinary AR pet system has the following limitations.

• The AR pet is pre-set and generally there is only one type. It cannot be created by users as they like.

• The interaction between the AR pet and human is set and cannot be triggered randomly.

• The whole system is more gamified and has less interaction with the actual scene. The whole system mainly embodies the function of the companion of pets. The whole system mainly embodies the function of the companion of pets. The system has no other function and lacks practicability.

Due to these limitations, the current AR pet system can't provide real pet raising experience for users, thus it can't well reflect the positive influence of pets on users.

Therefore, our system is devoted to solving this problem. Our system allows users to create AR pets according to their preferences. And according to the object recognition technology, AR pets can spontaneously interact with the recognized objects. In addition, AR navigation has been added to our system. Let AR pets be your Home Guide, meeting the practical needs of many users, especially the elderly users with memory impairment.

1.2 Organization of the thesis

The rest of the thesis is organized as follows: Chapter 2 will introduce the background and starting point of our research. Chapter 3 will present the research objectives and briefly outline our research methods. Chapter 4 will introduce our system design in detail. The design concept and method of each function are also introduced in detail. Chapter 5 is the system implementation part where the detailed environment and implementation will be talked about. Chapter 6 will introduce the related work of our research. Chapter 7 will be about the preliminary evaluation, we will compare our system with other similar systems and invite participants to carry out experimental investigations to verify the effectiveness and efficiency of our system. Chapter 8, will be the conclusion and future work part, where we will conclude the previous content and talk about future possibilities.

Chapter 2

Background

2.1 Augmented Reality

2.1.1 Definition

In Merriam-Webster dictionary, Augmented Reality means an enhanced version of reality created by the use of technology to overlay digital information on an image of something being viewed through a device (such as a smartphone camera). It combines virtual information with the real world.

The discussion of AR in academia can be traced back to the research of Mligram and kishino. Milgram and Kishino describe AR as a technology that "allows users to see the real world, with virtual objects superimposed on or mixed with the real world. [2]" And they also distinguish Augmented Reality from Mixed Reality and Augmented Virtuality, like Figure 2.1 shows.

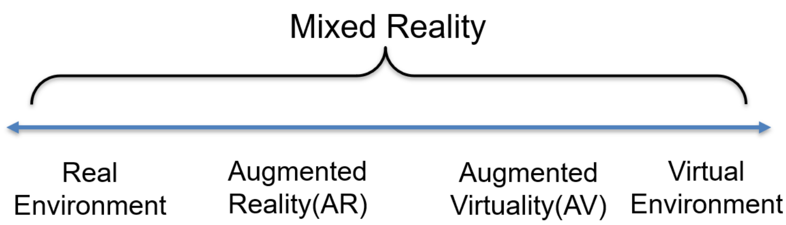


Fig. 2.1 Milgram's reality–virtuality continuum.

In this figure, we can see Miligram considers Augmented Reality(AR) belongs to Mixed Reality. Mixed Reality is a technology to merge real and virtual worlds. Augmented Reality (AR) is a subcategory of mixed reality that refers to all situations in which the display of the original real environment is enhanced through virtual objects. In contrast to it is Augmented Virtuality(AV) which is a technology that refers to the merging of real-world objects into virtual worlds.

Azuma et al. [3] further proposed a set of features that define AR. The characteristics are:

1. Combines real and virtual objects in a real environment;
2. Runs interactively and in real-time;
3. Registers (aligns) real and virtual objects with each other.

In recent years, some researchers have tried to classify different types of Augmented Reality technologies. However, there is no official statement. As Table 2.1 shows, Edwards-Stewart et al consider AR has six different types that fall under two overarching categories [4]. And a different type of augmented reality has different characteristics. But some experts consider AR could roughly be divided into three categories: marker-based AR, markerless AR, and GPS or location-based AR. In this research, we focus on the dynamic augmentation, which is also a kind of markerless AR.

Table 2.1 Summary of augmented reality categories and types.

Category	**Type**	**Characteristics**
Triggered	1.Marker-based	AR triggered by a visual marker.
	2.Location-based	AR triggered by GPS.
	3.Dynamic Augmentation	Responsive to the view of the object as it changes.
	4.Complex Augmentation	A combination of Marker/Location-based AR and Dynamic Augmentation.
View-Based	5.Indirect Augmentation	Augments a static view of the world.
	6.Non-specific Digital Augmentation	Augments a dynamic view of the world.

2.1.2 Markerless AR System

Actually, there is no official definition of markerless AR and marker-based AR. But we can find the most obvious difference between marker-based AR and markerless AR is whether a marker is needed or not. Marker-based AR has to use a marker as a trigger while markerless AR can be used without markers.

Markerless Augmented Reality (Markerless AR) is a term used to denote an Augmented Reality application that does not need prior knowledge of a user's environment to overlay 3D content into a scene and hold it to a fixed point in space [5]. Markerless Augmented Reality relies on the natural features of things rather than identifying specific markers. Compare with marker-based AR, markerless AR's most obvious advantage is flexible.

2.2 Current Pet Mode

Currently, there are four main pet modes: physical pet mode, pet experience hall, digital pet game and augmented reality pet system.

2.2.1 Physical Pet Mode

Today more and more people choose to keep pets to meet their emotional needs, especially the senior person. According to business weekly Shukan Toyo Keizai, 20% of Japanese pet owners are between 50 and 59 years old, and 16% are between 60 and 69 years old. The 5 best animals which are popular as pets in Japan are dog, cat, fish, bird, and rabbit.

However, the physical pet mode brings a lot of happiness to owners and also has a lot of bad influences. The first problem is that it will bring economic pressure to the owner. According to Pet Food Association of Japan(PFAJ) research, based on the data of 2014, the average life span of Japanese pet dogs is 14.25 years old, and their owners need to spend about 1.19 million yen in their lifetime, while the average life span of pet cats is 14.56 years old, and their owners need to spend about 700,000 yen for them.

When it comes to senior persons, the negative aspects of pet ownership were more obvious, including the practical and emotional burden of pet ownership and the psychological impact that losing a pet has. The most common trouble is pets will bring a safety challenge to the elderly. Some pets are large and less obedient. Elderly raising them can be troublesome and prone to danger. Owners can fall after being pushed by larger pets, or they may lose their balance when leaning down to feed a pet [6]. Pet care also brings many responsibilities-walking, feeding, washing, regular play, and care. These are all very stressful for the elderly. What's more, the death of pets during the breeding process causes psychological shock to the elderly. And compared with young people, this sadness will last longer and have a greater impact on the elderly.

2.2.2 Pet Experience Hall

Pet experience hall is a type of shop that can provide people with the opportunity to contact with different types of pets. From the most common cats and dogs to rabbit hamsters, there are many types of pets. The most common and famous one is the cat cafe.

Although such facilities allow busy citizens to enjoy pets, they also bring many hidden dangers. First of all, when pets face a large number of people, they will make them nervous, and sometimes they may even behave out of control. Secondly, people cannot establish long-term stable contact with pets in such facilities. This connection is short-lived, so the emotional maintenance between pets and people is also fragile.

2.2.3 Digital Pet Game

The Pet-themed game is very common. Digital pet game allows users to raise pets virtually. It is very convenient and does not require the user to spend too much energy and financial resources.

For example, as Figure 2.2 shows, Tamagotchi is a very famous handheld digital pet. Most Tamagotchi is housed in a small egg-shaped computer with an interface consisting of three buttons. Therefore, the interaction between the user and the virtual pet is fixed and relatively simple.

Fig. 2.2 Tamagotchi

2.2.4 Augmented reality Pet System

The AR pet system becomes more and more popular. Based on the AR technology, the AR pet system can give you a more real experience. There are two main types of AR pet systems. The first type of AR pet system is based on the AR marker. The user recognizes the marker by scanning to call out the AR pet. And then the system allows users to do some interactions with the AR pet. The other AR pet system is markerless. Users can get their AR pets when they open the system. But the user interaction of these AR systems is similar. The user controls the AR pet by clicking the button. In that case, the user cannot enjoy the interaction with pets more immersively and smoothly.

Fig. 2.3 The current AR pet system

Chapter 3

Research Goal Covers User Cases and Approach

3.1 Goal

The fundamental goal of our research is to **improve the existing augmented reality pet system to make augmented reality pets serve as an emotional supporter and home guides.**

The goal can be divided into the following points:

1. System allows users to created their AR pets. Users can upload any photos of pets they want to our system. Our system will generate a 3D AR model based on these photos.

2. System allows AR pets to interact with users and real-world objects in real-time and dynamically. The interactions contains gesture control and object detection.

3. System provides an AR pet navigation function. The AR pet will guide you home efficiently and accurately.

3.2 Use Cases

Our system has two user cases around the AR pet. The first use case is the interaction part in which the AR pet serves as an emotional supporter. In the second use case, our system provides AR navigation. The AR pet will perform as a guider to guide you back home.

3.2.1 Use Case 1

A user wants to company with a pet though he doesn't have a pet.

Then he can upload a series of pictures of what pet he wants to own to our system. And our system will generate a 3D AR pet for him. Once pass the face detection, the AR pet belongs to him will appear on the screen. After that, the user can fully enjoy the fun and convenience brought by the AR pet. The user can do real-time interaction with his AR pet. For example, he can control the AR pet's size, location, and rotation by the gesture. And the AR pet automatically triggers interactions according to the object after the user scans real-world objects.

(a) Gesture control (b) Object detection

Fig. 3.1 Real-time interactions

3.2.2 Use Case 2

Many elderly people with poor memory, especially those with Alzheimer's disease, often forget the way home. Our system can help this kind of people. Once the user goes out, he can open our system and scan the road and click the screen to add anchor points. Our system will detect the plane and store anchor points. When he wants to go back home, he only needs to click the "Back Home" button, and then his AR pet will appear and guide him to go back home through the stored anchor points.

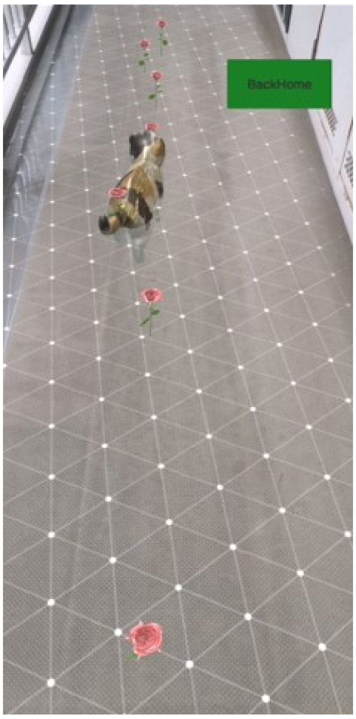


Fig. 3.2 AR pet home guide function

3.3 Approach

To achieve the goal, we will introduce our approach to the following aspects. Firstly, 3 key aspects will be introduced: 3D model construction, real-time interaction, and AR pet home guide.

In order to generate the 3D AR pet, we generate a created 3D mesh based on users' overlapping photos. However, the current pet model is static. In order to make it move like a real pet, we need to build the bone structure for it.

After we get the AR pet model, we design several application scenes for it. The first application scene is real-time interaction.

AR pet's real-time interaction based on object detection. Our research uses TensorFlow to do object detection. So, first all, we should train our model. At first, we should collect a lot of pictures. Generally, 50 pictures for one category would be sufficient. After preparing the images, we need to label them manually. We use LabelImg for this part's work. It saves label files in Pattern Analysis, Statistical Modelling and Computational Learning(Pascal) Visual Object Classes(VOC) format. The Label map also needs to be created. Then, we need to convert our data into the TFRecord format which is a simple format for storing a sequence of binary records. Because TFRecord is an important data format designed for Tensorflow. Finally, we can train our model and export it to use. We combine the object detection with AR pet's activity to realize AR pet's real-time interaction. The AR pet will trigger different actions based on different object detection results. Figure 3.3 shows the framework of this part.

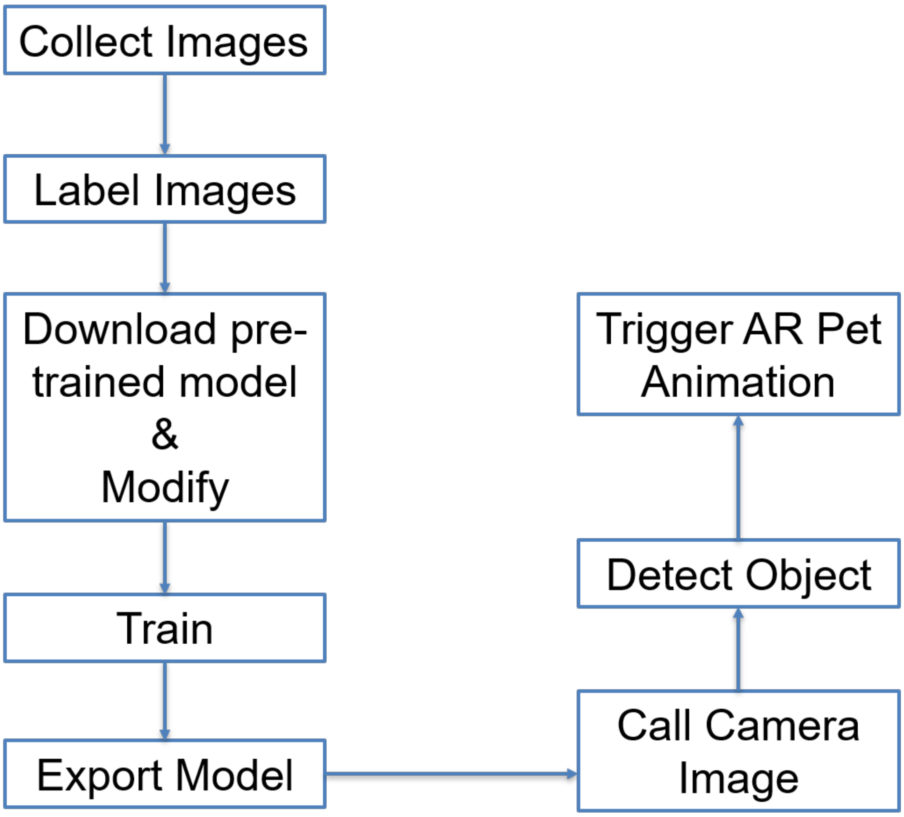


Fig. 3.3 Real-time interaction framework

To achieve the AR pet home guide function, we import ARCore to detect the plane. And touch the screen to add anchor on the detected plane. We will use a script to store this anchor's position in a list. This list will help our system to design the path. When the user clicks the "Back Home" button, the AR pet will appear and guide the user back home along with the anchors. Figure 3.4 shows how the home guide part works.

Fig. 3.4 AR pet home guide framework

To verify our approach achieves our goal and prove the effectiveness of our approach, we will perform several experiments.

Chapter 4

System Design

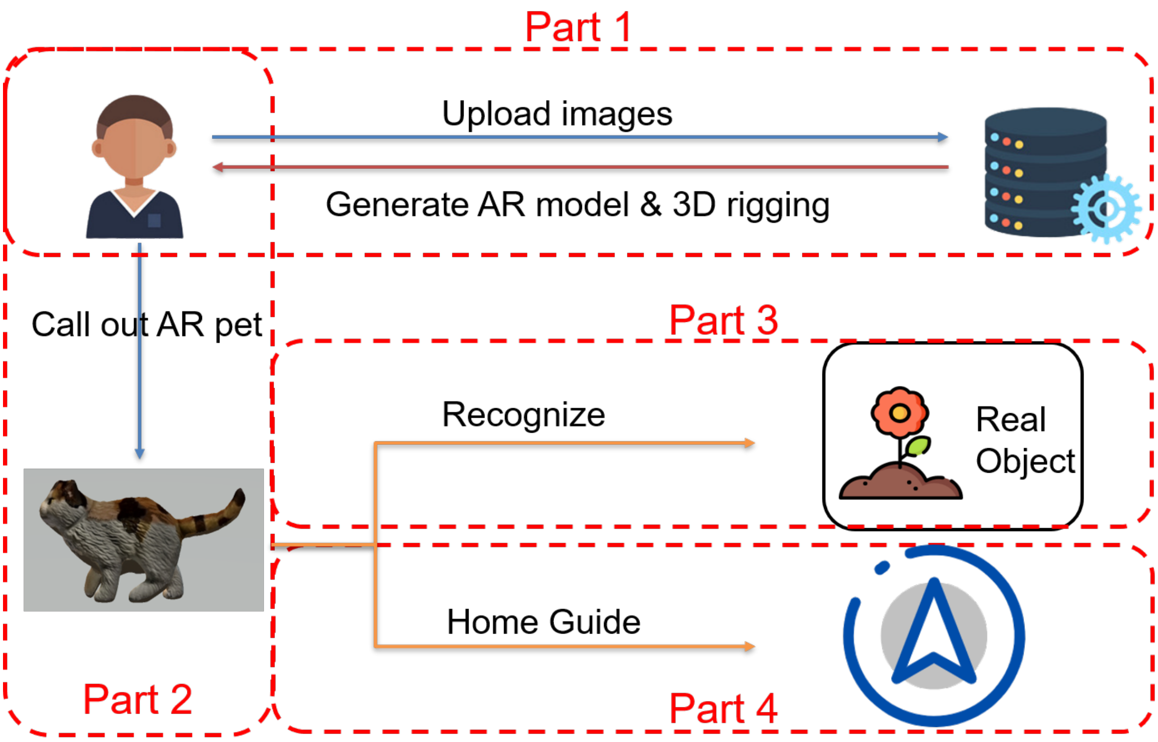


Fig. 4.1 Overview of our system

In this chapter, we will introduce our system design and each point of our approach. Figure 4.1 shows the overall structure of the system, and we will divide the system description into four parts:

- Part 1 is about the 3D model construction;
- Part 2 talks about user authentication;
- Part 3 is the real-time interaction part. Data training, object recognition will be introduced;
- Part 4 introduces the AR pet home guide function.

4.1 3D Model Construction

In our proposed system, we generate the 3D pet model based on the user's photos. The user should take a photo of about every 20 degrees with overlapping content between photos to construct a 3D model.

In this study, as shown in Figure 4.2, we took 74 photos of a kitten model toy at different angles to build a 3D cat model. And then we can created the AR pet model with the help of the Reality Capture API. The Reality Capture API provides a set of endpoints for the Photo to 3D capability. These endpoints allow the user to manage the process of generating a 3D mesh from overlapping photos.

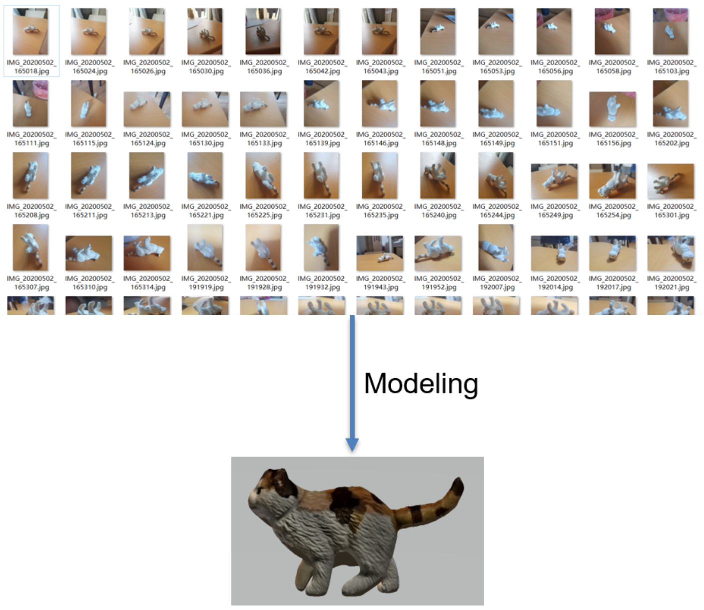
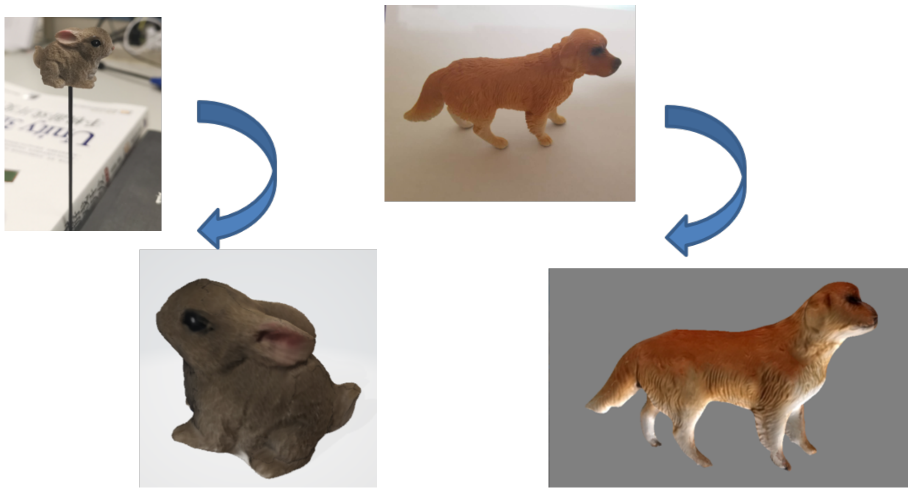


Fig. 4.2 3D Modeling based on photos

Using the same method, we constructed two other 3D models, a rabbit and a dog, as shown in Figure 4.3. In our system, we use the cat model as example to design application scenarios. But in actual operation, users can interact with any AR pets generated from the pictures they upload.



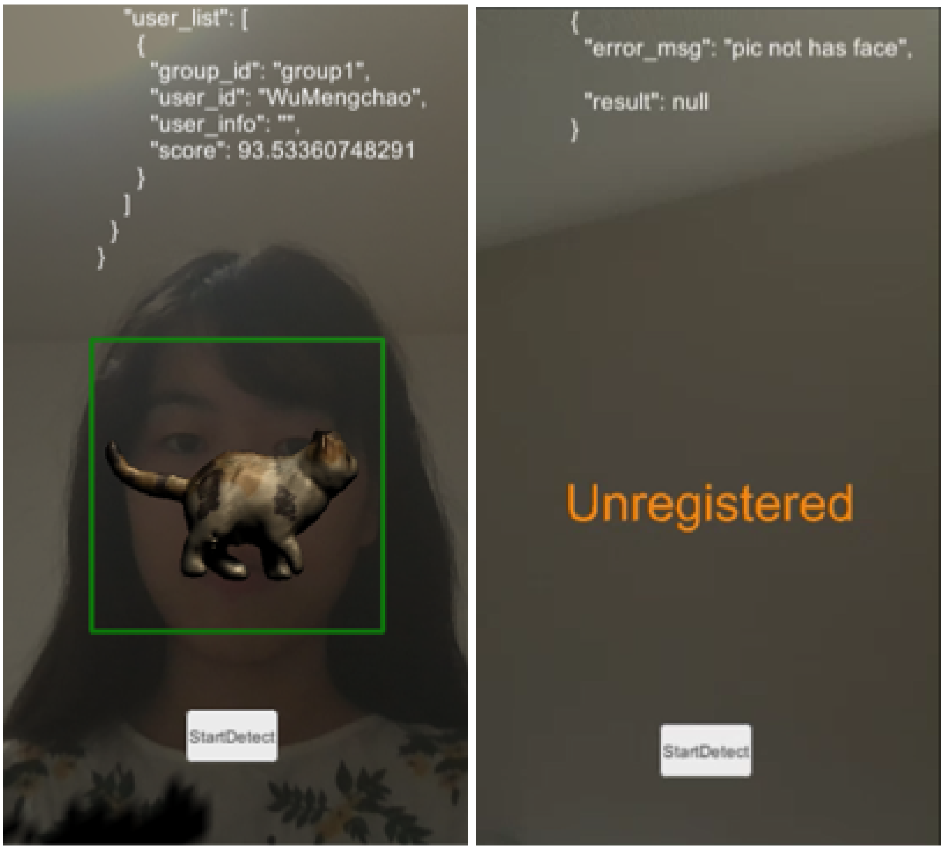
(a) Rabbit model (b) Dog model

Fig. 4.3 3D pet models.

4.2 User Authentication

To build a connection with the user and his AR pet, we add a user authentication part. The user looks at the camera and clicks the "Start Detect" button. Face recognition will be performed on the user.

User authentication is a face authentication step for whether the current user is a registered user. Figure 4.4 shows two different user authentication results.



(a) Successful user authentication (b) Failed user authentication

Fig. 4.4 User Authentication

The face authentication system of our system calls the Baidu API and creates a face recognition library in the cloud. When our system performs face recognition, the page will display the percentage score of the similarity between the user and the face pictures in the face recognition library.

When the score is more than 90, we recognize this user is the person who has registered in our system. The reason for choosing the score of 90 is because our repeated attempts have shown that as long as it is the same person, even if he changes his/her hairstyle, his/her authentication score will be higher than 90. Once facial recognition is successful, the AR pet belonging to the user will appear on the screen as Figure 4.4(a) shows. If facial recognition is not successful which means the score less than 90 or our system cannot recognize the user, an error message will appear like Figure 4.4(b) shows. If the authentication fails, our system also provides the opportunity for re-authentication. The user can click "Start Detect" again to authenticate.

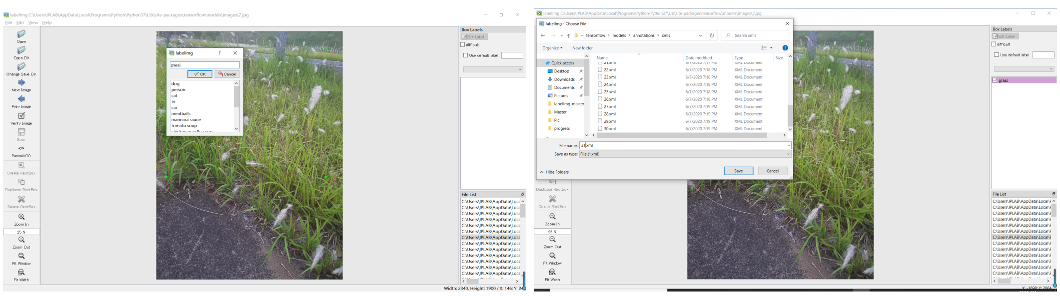
4.3 Real-time Interaction

4.3.1 Data Training

Before we perform object recognition, we need to have the model file and the corresponding label file. Our system relies on TensorFlow for object recognition. Currently, TensorFlow has several trained object detection models that can be used in Unity. But the types and number of models are limited. If you need to recognize a specific object, you still need our own training data.

First, we should collect a large number of images for custom object detection. We can get images through video screenshots or download pictures in batches online. In order to be able to accurately identify a type of object, we need about 50 different jpg format images of the type of object.

After getting the image, we need to label the image one by one. In our research, we choose labelImg as a tool to do this part work as Figure 4.5. This software will save the label file after you labeled the image. So, every image has a corresponding label file. We should sort these label name and the corresponding id in one label map file.



(a) Label the image in labelImg (b) Save the label file

Fig. 4.5 Example of labeled grass in labelImg

Then, we need to download a pre-trained object detection model. Because we need to use its checkpoint. So that we can train our custom data set. In this study, we download ssd_mobilenet_v1_coco and modify its Config file to adapt our data set.

When the preparation work is ready, we can start training our model. We observe the model with Tensorboard. Figure 4.6 shows our model's TensorFlow graph. Then we should export our model to be used for inference.

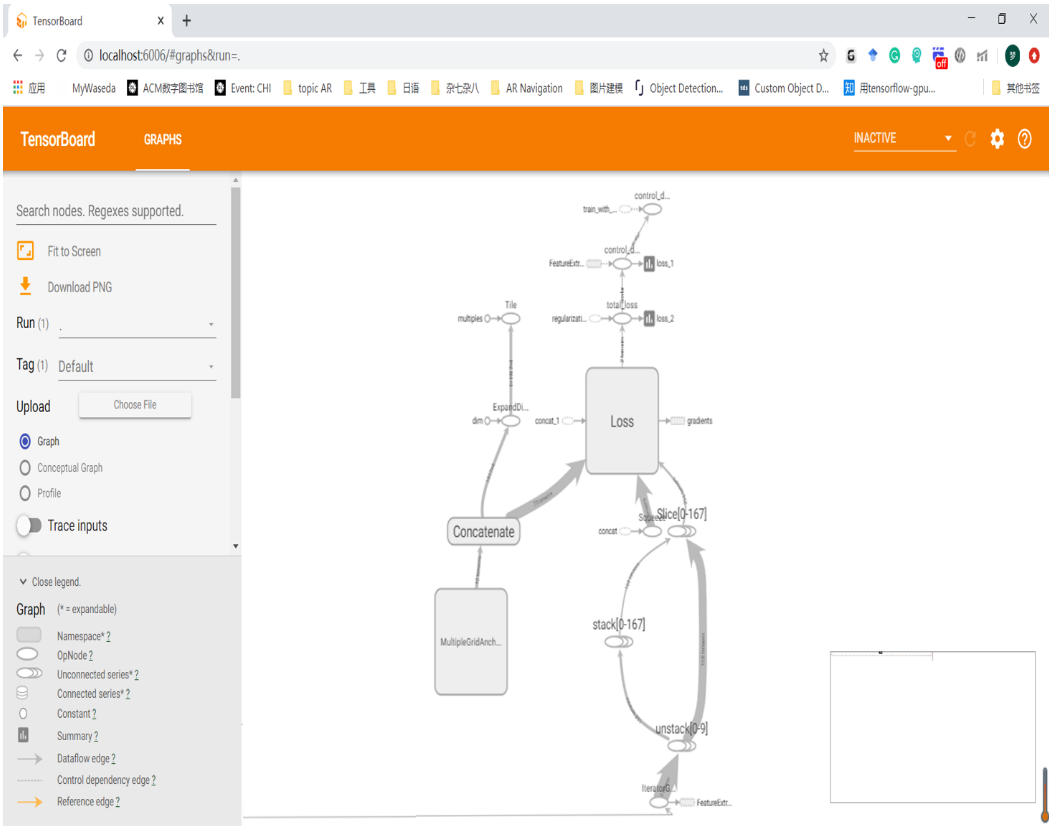


Fig. 4.6 The TensorFlow Graph of the model

4.3.2 Object Recognition

After we train our created detector, we can use it to detect objects shows in the camera. At the same time, our system also allows AR pets to interact based on object recognition results. Figure 4.7 shows the interface of this part.

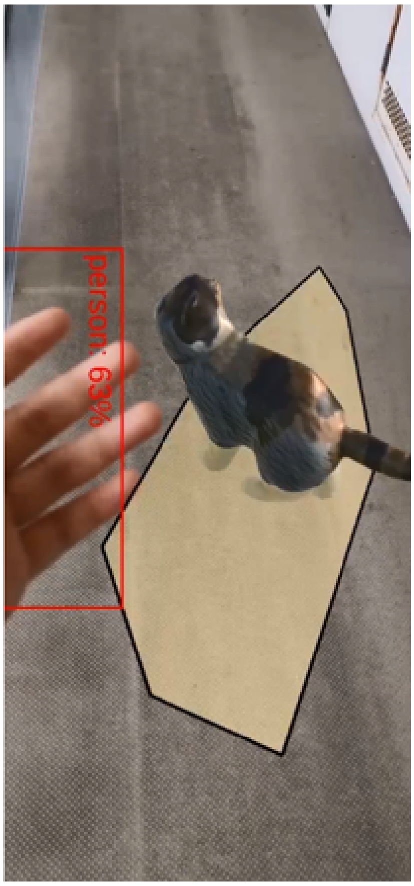


Fig. 4.7 Object recognition and AR pet interaction

In order to achieve this function, we did the following work:

- Our research refers to ARFoundation to get the camera frame. We attempt to get the latest camera image by calling the XRCameraImage class in ARFoundation. And then we convert the camera image to an RGBA texture and draw it on the screen. Meanwhile, we should set the RawImage's texture as the camera image so that we can visualize it.
- Every time a new camera frame is captured, we pass the processed texture to the trained detector and run inference to get the detected objects. The output will be a box area containing the object and the corresponding label name.
- When we put an AR pet in the same scene, our system uses ARfoundation to realize the adaptation of the AR pet. We first use ARFoundation to identify the plane and

then place the AR pet on the identified plane. Ensure that AR pets are suitable for the environment. Uncommon sights such as pets floating in the air will not appear.

- Then, we will loop the outputs and look for the target object according to their label names. When our system finds the target object, the AR pet's animation will be triggered. In this way, real-time interaction between the AR pet and the real scene objects is realized.

4.4 Gesture Control

Our system also allows users to interact with the AR pet through simply gesture. Once our system detects a available plane, an AR pet will put in that plane automatically. Then users can do interaction with that AR pet. For example, users can zoom in and out of the size of the AR pet with two fingers. They can also drag the AR pet to select any location in the detected plane to place the AR pet. In order to view the AR pet conveniently and clearly, our system also provides users with the ability to rotate the AR pet. The user can rotate the AR pet's angle by sliding the AR pet on the screen clockwise or counterclockwise with one finger. Figure 4.8(a) shows zooming in and zoom out the AR pet, Figure 4.8(b) shows rotating the AR pet.

(a) Zoom in and zoom out the AR pet (b) Rotate the AR pet

Fig. 4.8 Example of gesture control

If the user repositions the AR pet to a farther position on the detected plane by tapping the screen, our AR pet will slowly come to you. This feature is implemented through animations and scripts in the Unity editor, and users do not need to perform any operations. Figure 4.9 is a screenshot of the AR pet traveling.

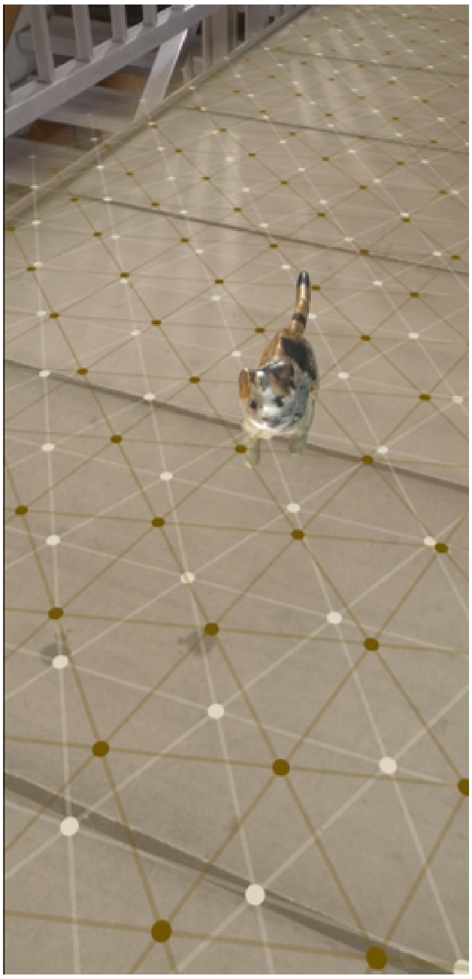


Fig. 4.9 A screenshot of the AR pet traveling

4.5 AR Pet Home Guide

AR pet home guide is another part of our system. In this part, our system will achieve the AR navigation and the guider will be your AR pet. Kuwahara et al. [7] has confirmed that route navigation by the AR character agent makes route identification easy.

The AR navigation of our system aims to help users with poor memory, especially the elderly with Alzheimer's disease, find their way home.

The structure of the AR pet home guide can be divided into three parts: Detect plane, add anchors, and AR navigation.

4.5.1 Detect Plane

When the user starts to use the AR navigation function of our system, the first thing he needs to do is to scan the road under his feet with the camera of the mobile phone.

As Figure 4.10 shows, our system refers to ARCore to detect surfaces of the real world. Generally, the plane you detected is the road you need navigation. The detected plane has its position and size. After detection, the detected plane can be displayed and utilized.

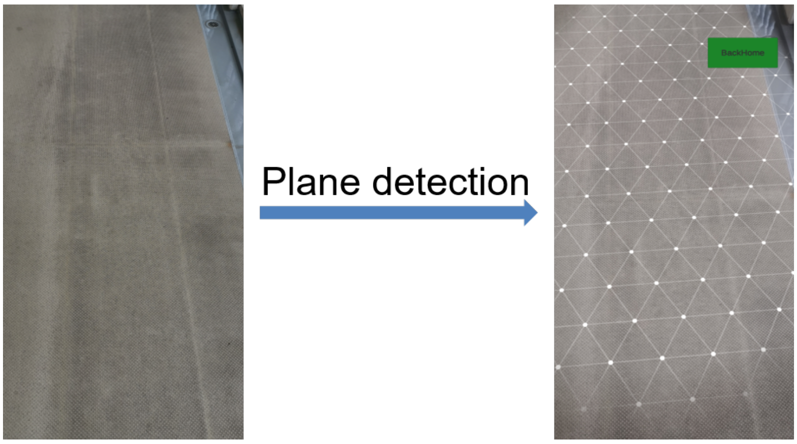


Fig. 4.10 The process of plane detection

The detected plane will serve as a surface for adding anchors and navigation.

4.5.2 Add Anchors

When the user starts to move, the user can touch the system screen from time to time to set an anchor point. Anchors play a role in dividing the entire journey into sections in our system. Users can add anchors on the detected plane. These anchors can help our system to navigate more accurately and efficiently. Figure 4.11 shows the process of add anchors.

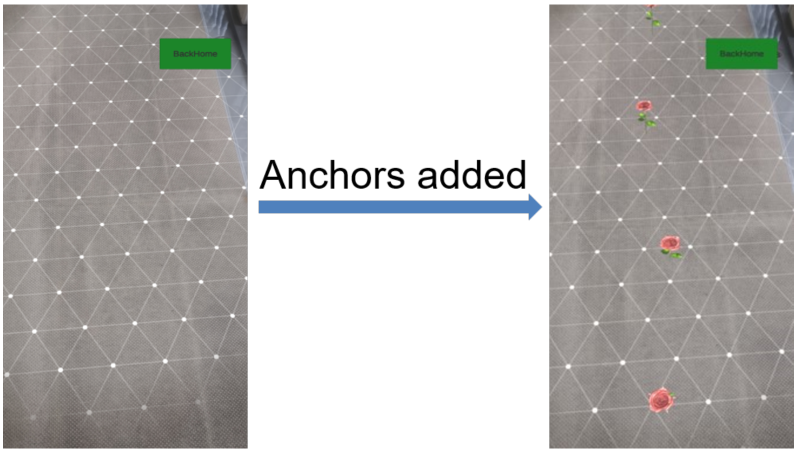


Fig. 4.11 The process of add anchors

Our system adds the anchor-based on where the user hits. We record the location and rotation where the user hits and maps this information to the corresponding location on the detected plane. Then an anchor will be generated in the corresponding position and direction on the plane.

Our system will store the information of these anchors such as position, direction, and order in a list. After that, our system will navigate based on this information.

4.5.3 AR Navigation

When the user finishes going out and wants to go home but can't remember the way back, he can use our navigation system for AR navigation home.

Our system provides users with a one-click navigation function. When the user clicks the "Back Home" button on the screen, this user's AR pet will appear in his current location. Then the pet will navigate for the user along the sequence from the last set anchor point to the first set anchor point which is previously set by the user. Figure 4.12 shows the AR navigation interface of our system.

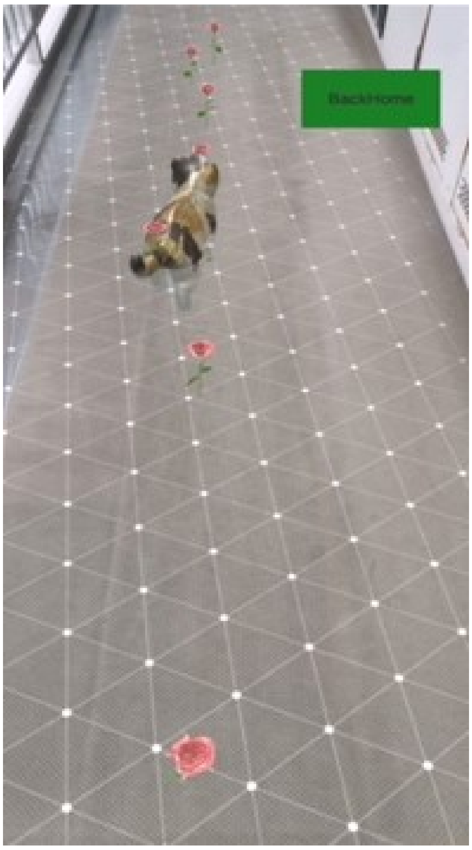


Fig. 4.12 The AR navigation interface

According to a study of TranSafety Incorporation [8], the average walking speed of the elderly is 3.2 km/h to 3.9 km/h. For elderly people with poor memory or Alzheimer's disease, the walking speed will be slower. Therefore, our system sets the AR pet's navigation speed to 3km/h which is very friendly for the elderly.

Chapter 5

System Implementation

5.1 System Hardware

We divided our system into two parts: Smartphone users and the server. To achieve our system, we need some hardware:

- A PC to support the programming environment;
- A smartphone with a camera to realize our system.

Our system uses a PC with the Windows 10 operating system to provide a platform for the system to process programs. Table 5.1 shows the information on our PC.

Table 5.1 The information of PC.

Category	**Information**
Operation System	Microsoft Windows 10
CPU	Intel(R) Core(TM) i7-6500U CPU @2.5GHz 2.59GHz
Ram	8.00GB

Our system needs to recognize the environment and detect objects. So, we need a mobile phone with a camera to achieve this goal. Meanwhile, we develop our system on the Android platform. We'd better choose an Android smartphone to use our system. So we use Huawei Mate 20 Pro smartphone as the device as Figure 5.1 shows.

Fig. 5.1 Huawei Mate 20 pro smartphone.

5.2 Development Environment

To develop our system, we need some plug-ins and technical supports as follows:

1. The Reality Capture API which provides a set of endpoints for the Photo to 3D capability. We use it for the 3D model construction;
2. The Baidu API which can compare the current face with the face in the cloud face library. We call it for the face recognition;
3. ML-Agents, a Unity Machine Learning Agents Toolkit, which allows users to use Unity as a platform to train and embed intelligent agents using the latest advancements in machine learning.
4. TensorFlowSharp plug-in which offers the possibility to use pre-trained TensorFlow graphs inside of the game engine.
5. ARcore and ARFoudation both are good support for Augmented Reality.

We build our system in Unity 2019.2.0f1 version and use C as the programming language. We also used 3DMax to process our 3D model.

5.3 Framework

Figure 5.2 shows the framework of our system. First, the user takes some photos of his favorite pet and uploads these photos to our system. The system will upload these photos to the Reality Capture API URL to generate a 3D pet model. To call out this model as an AR pet, this system will perform user authentication on the user. Our system will capture the user's face, and upload this picture to Baidu API URL for comparison. If the user authentication is successful, the AR pet will be displayed in the interface. Also, the AR information in the interface will change with the user's different interactions.

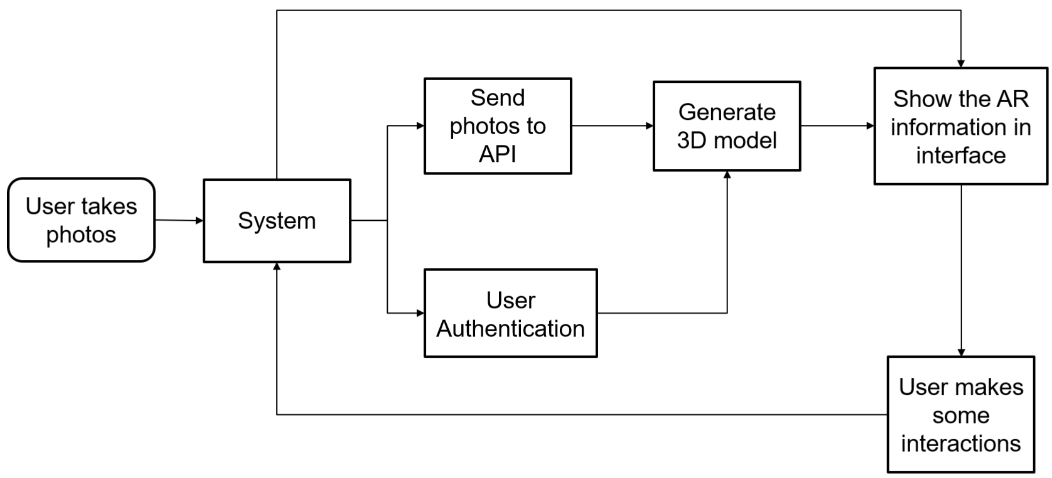
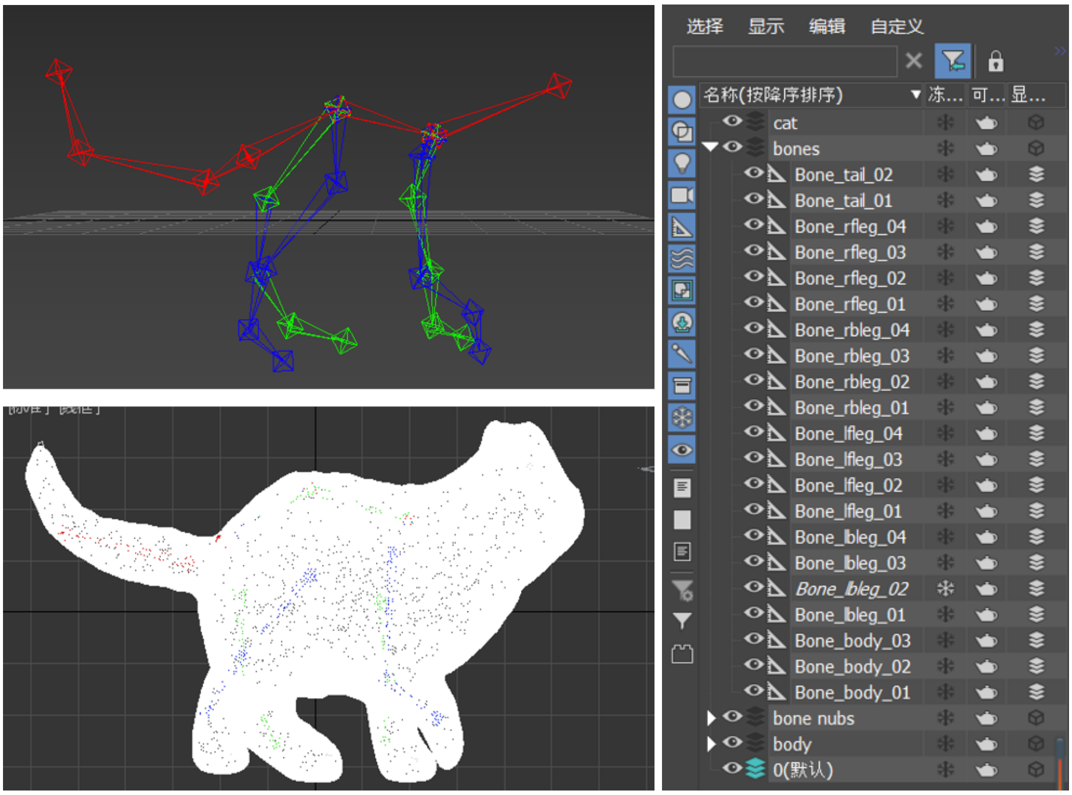


Fig. 5.2 System diagram.

5.4 3D Model Rigging

While when we get the 3D model is static after calling the Reality Capture API. It cannot move and do actions. We should rig it. 3D rigging means create the bone structure of the 3D model to allow the object to be animated freely. We rig our model in 3DMax. We constructed a series of bones representing the skeletal structure. These bones can be transformed using tools in 3DMax which means their position, rotation, and scale can be changed. Animation can be recorded by recording these aspects of bones along the timeline.



(a) Rig the model (b) Bone structure of the model

Fig. 5.3 3D Rigging

Figure 5.3 shows the bone structure of the cat model. We divided into five parts to build a skeletal system of this model. These are the legs, body torso, tail, head, and feet. Now this cat model can do animation freely.

5.5 Environment Recognition

After we get an AR pet, we need to detect a platform to put our AR pet. In that case, we should do environment recognition work. We use ARCore to do that work.

ARCore is a software development kit developed by Google that can be used to build augmented reality applications. It has a strong environmental understanding ability.

To use ARCore in our system, we should download ARCore SDK for Unity and import it in our project. And then we should configure project settings to adapt ARCore.

After finishing the preparation work, We will use the point cloud data generated by the ARCore camera to detect planes. At the same time, we will also visualize the detected planes to help users indicate where an available plane is.

Firstly, we should create a new plane. When detecting a plane in the real world, we need a way to represent this feature in the virtual space, which is to use the visualized virtual plane. Then, we also need to assign a material to the plane so that the user can see the plane after instantiation. Thirdly, we should attach the DetectedPlaneVisualizer script that comes with ARCore to our plane. This script helps us to render the detected or extended plane.

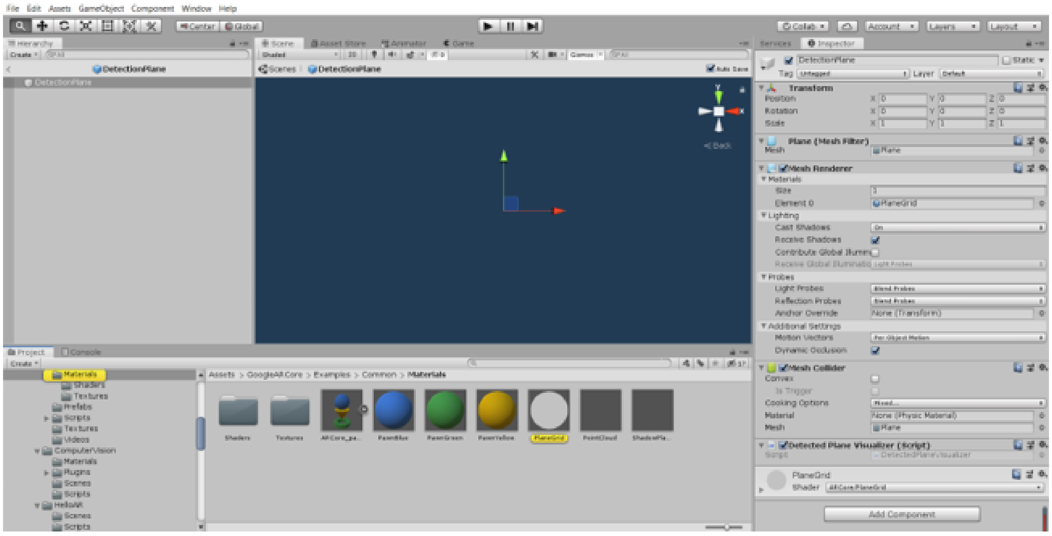


Fig. 5.4 The configuration of the plane.

We also should write codes to visualize the detected plane successfully. Figure 5.5 shows the codes that iterate over planes found in this frame and instantiate our visualized virtual plane which is created in unity to visualize them.

```
public void Update()
{
    // Check that motion tracking is tracking.
    if (Session.Status != SessionStatus.Tracking)
    {
        return;
    }

    Session.GetTrackables<DetectedPlane>(m_NewPlanes, TrackableQueryFilter.New);
    for (int i = 0; i < m_NewPlanes.Count; i++)
    {

        GameObject planeObject =
            Instantiate(DetectedPlanePrefab, Vector3.zero, Quaternion.identity, transform);
        planeObject.GetComponent<DetectedPlaneVisualizer>().Initialize(m_NewPlanes[i]);
    }
}
```

Fig. 5.5 Visualize detected planes.

Now, we get a platform to put our AR pets. How to place the AR pet on the three-dimensional detection plane with the operation on the two-dimensional mobile phone screen needs the help of ARCore. ARCore's approach is to do Raycast. The basic idea of Raycast is to emit an infinite ray line from a point in one direction in the three-dimensional world. In the ray direction, once colliding with the model added with the collider, an object detected by the collision will be generated. We can use Raycast to detect the location of the collision.

```
if (Frame.Raycast(touch.position.x, touch.position.y, raycastFilter, out hit))
{
    if ((hit.Trackable is DetectedPlane) && Vector3.Dot(FirstPersonCamera.transform.position - hit.Pose.position, hit.Pose.rotation * Vector3.up) < 0)
    {
        Debug.Log("Ray hit the back of DetectedPlane! ");
    }
    else
    {
        var prefabIns = Instantiate(prefab, hit.Pose.position, hit.Pose.rotation);
        prefabIns.transform.Rotate(0, mModelRotation, 0, Space.Self);
        var anchor = hit.Trackable.CreateAnchor(hit.Pose);
        prefabIns.transform.parent = anchor.transform;
    }
}
```

Fig. 5.6 Put AR pet in the platform using Raycast.

As Figure 5.6 shows, in our system, we use Raycast to detect the location of the collision and instantiate the AR pet in this position. At the same time, an anchor is generated and the AR pet is mounted on the anchor so that ARCore can track the position of the AR pet.

5.6 Put the AR pet into the real environment

We allow users put the AR pet in real environment and users can control the AR pet with different gestures. The gesture control part is totally achieved by script. We detect the number of touches on the screen and give different touch counts different meanings. If an AR pet already exists in the scene and the number of touches is one, then we think that the user wants to rotate the AR pet. If the number of touches is two, then we think that the user wants to control the size of the AR pet.

We also design a scene that the AR pet come to the user slowly. To achieve that, we should make Keyframe animations for the pet model based on the established bone structure. We do this work in the 3DMax in advance. Then we can import the animation resources into Unity and apply them to the scene.

5.7 Object Detection

We design our system in Unity3D, so, we should import some plug-ins to support TensorFlow algorithms in Unity3D.

Unity Machine Learning Agents Toolkit(ML-Agents) is an open-source project which enables researchers and developers to create simulated environments using the Unity Editor and interact with them via a Python API [9]. ML-Agents contains three advanced components:

1. Learning Environment: It contains Unity scenes and all game characters.
2. Python API: It contains all machine learning algorithms used for training. The Python API is not part of Unity, but is external and communicates with Unity through External Communicator.
3. External Communicator: It connects Unity environment with Python API. It is located in the Unity environment.

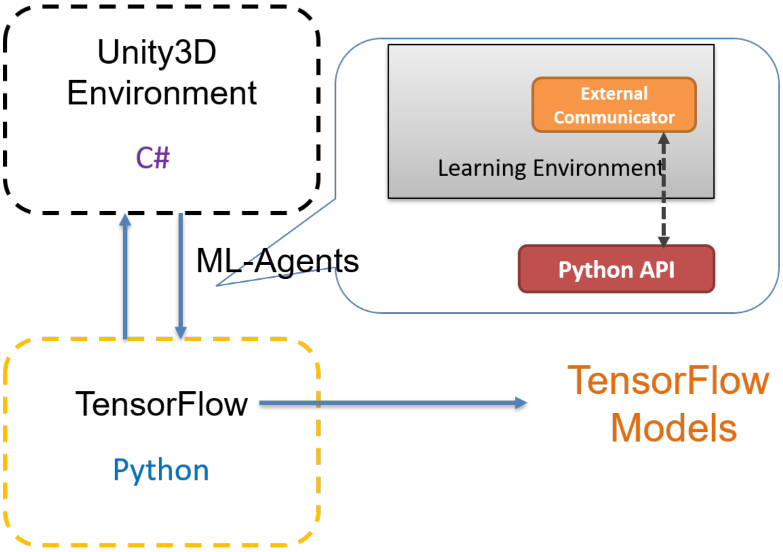


Fig. 5.7 The framework of ML-Agents.

Like Figure 5.7 shows, ML-Agents provides a possibility to refer TensorFlow to do object detection in Unity3D. But to achieve the conversion between C and python, we need to add the TensorFlowSharp plugin to our project. It allows us to perform TensorFlow operations (a Python API) in C. As we all know, Unity3D uses C as its default language.

TensorFlowSharp is another plug-in we import in our system. It is a package of the TensorFlow C language interface, which is convenient for C developers to use TensorFlow in the project.

In our system, we need to detect the objects in the picture captured by the mobile phone camera. So, we should get the camera frame and be able to process it. In that case, we refer ARFoundation to get the AR camera picture, like Figure 5.8 shows. We attempt to get the latest camera image and apply the updated texture data to our texture.

```
XRCameraImage image;
if (!cameraManager.TryGetLatestImage(out image))
{
    return;
}

var format = TextureFormat.RGBA32;

if (m_Texture == null || m_Texture.width != image.width || m_Texture.height != image.height)
{
    m_Texture = new Texture2D(image.width, image.height, format, false);
}

var conversionParams = new XRCameraImageConversionParams(image, format, CameraImageTransformation.None);
var rawTextureData = m_Texture.GetRawTextureData<byte>();
try
{
    image.Convert(conversionParams, new IntPtr(rawTextureData.GetUnsafePtr()), rawTextureData.Length);
}
finally
{
    image.Dispose();
}
defaultBackground = rawimage.texture;

this.rawimage.texture = m_Texture;

this.backgroundSize = new Vector2(this.m_Texture.width, this.m_Texture.height);
m_Texture.Apply();
```

Fig. 5.8 Capture camera image.

Then, we will pass the processed graph we captured to the TensorFlow detector model and do detection. Generally, the model file is a .byte file. We should load this model file and its label file. And then we can run it to get a detection result. Figure 5.9 shows the object detection procedure.

```
private void LoadDetector()
{
    this.detector = new Detector(
        this.modelFile.bytes,
        Regex.Split(this.labelsFile.text, "\n|\r|\r\n")
            .Where(s => !String.IsNullOrEmpty(s)).ToArray(),
        detectImageSize);
}
```

(a) Load the model

```
private async void TFDetect()
{
    UpdateBackgroundOrigin();

    var snap = TakeTextureSnap();
    var scaled = Scale(snap, detectImageSize);
    var rotated = await RotateAsync(scaled.GetPixels32(), scaled.width, scaled.height);
    this.boxOutlines = await this.detector.DetectAsync(rotated);

    Destroy(snap);
    Destroy(scaled);
}
```

(b) Start detection.

Fig. 5.9 Object detection

If the detection runs successfully, a rectangle enclosing the detected object will be displayed on the screen, and the label of the detected object will also be displayed.

5.8 Path Organization

In the AR pet navigation part, we use AR pet as a guider to guide users back home. So, the AR pet should move base on an organized route.

In our system, this planned path is set by the user while walking out. Users will set anchor points along their walks. When the user clicks on the screen of the smartphone, an anchor point is generated at the clicked position. The shape and pattern of the anchor have been designed in advance.

```
placementPose = hit.Pose;
var FoxObject = Instantiate(routePoint, hit.Pose.position, hit.Pose.rotation);
FoxObject.transform.Rotate(0, mModelRotation, 0, Space.Self);
var anchor = hit.Trackable.CreateAnchor(hit.Pose);
FoxObject.transform.parent = anchor.transform;
wayPoints.Add(FoxObject);
```

Fig. 5.10 Set anchor points in the detected plane.

After obtaining the anchor points along the way, we need to store their location and direction information in the form of a list for future navigation.

When users press the "Back Home" button, the system start navigation. At first, our system will judge whether there already exists an AR pet guider in the interface. If there is no AR pet guider in the interface, the AR pet belongs to the user will appear in the last anchor point and start to guide the user.

```
public void Go()
{
    go = true;
    if (catVisible)
    { // already following

        Destroy(cloneCat);
    }

    cloneCat = GameObject.Instantiate(cat, wayPoints[wayPoints.Count - 1].transform.position, Quaternion.identity);
    catVisible = true;

}
```

(a) Call out the AR pet in the last anchor point

```
public void MoveToNextPoint()
{


    cloneCat.transform.LookAt(wayPoints[wayPoints.Count-num-1].transform.position);

    cloneCat.transform.position += cloneCat.transform.forward * speed * Time.deltaTime;

}
```

(b) Navigation at a certain speed.

Fig. 5.11 Call out AR pet and start navigation

Navigation will be from the last anchor point to the first anchor point set by the user, which is the user's starting point, usually the user's home. And the AR pet will pass through the set anchor points one by one in order, and finally, lead the user to find the way home. The speed of the AR pet can be adjusted, the default speed of our system is 3km/h.

Chapter 6

Related Work

Related works will be introduced in this session. There will be three kinds of related works, the first is about human-pet relationships, the second is the work about the pet system and the third is about AR navigation.

6.1 Human-pet Relationships

There are many researches focus on the emotional connection between pets and human beings. What emotional impact do pets have on humans? What is the degree of such influence? Different researchers have different opinions.

Sigal Zilcha-Mano et al. [10] concentrate on the dimensions of human-pet relationships. They used attachment theory as a framework to examine human-pet relationships. They designed a Pet Attachment Questionnaire (PAQ) to find that, just like attachment orientation in interpersonal relationships, pet attachment orientation consists of two orthogonal dimensions: attachment anxiety and avoidance.

There are also a number of studies interested in what influences pets will affect human beings. McConnell et al. [11] performed three studies to exam pets that can serve as important sources of social support. Their study 1 found pet owners fared better on several well-beings (e.g., greater self-esteem, more exercise) and individual-difference (e.g., greater conscientiousness, less fearful attachment) measures. And then they carried out the study 2 in a different community sample. They found the degree of the well-being of pet owners is positively related to the degree that pets meet social needs. Study3 proved that the companion of pets can reduce the negative emotions brought by lack of social intercourse by bringing pet owners and pets into the laboratory.

Enders-Slegers et al. [12] have confirmed that pet owners feel close to their pets and seek and enjoy this closeness. Pets can be used as primitive defense mechanisms, as "objects of identification" and as "balances" when the elderly lose relatives, friends and partners. Similarly, Francesca Moretti et al. [13] also confirmed the positive effect of pets on the elderly, especially the elderly in patients affected by dementia, depression, and psychosis. Allen Karen et al. [14] have even proved that pets are helpful to people's physical health. They found the heart rate and blood pressure levels of people with pets decreased significantly during the stationary baseline period. Compared with the baseline level, the increase (i.e., reactivity) during mental calculation and cold pressing was significantly smaller and recovered faster. Roswitha Scheibeck et al. [15] revealed that pets, especially dogs, have and indeed have multiple meanings for the elderly from a gerontological perspective by employing qualitative social research methods.

However, the positive effects of pets on human beings cannot be brought into full play. Because in many places where companion animals are most needed, such as hospitals, nursing homes, and educational institutions, companion animals are prohibited. For example, studies have shown that companion animals are of great benefit to the elderly, but most elderly people cannot keep pets due to their economic conditions or housing conditions [16]. Since the elderly do not have animals in public places and communities, the elderly find few opportunities to interact with companion animals [17]. This kind of situation does not only happen among the elderly. Some studies show that the frequency of interaction between people and pets is gradually decreasing [18].

At the same time, raising a pet also means responsibility and sometimes this responsibility will become a burden. June McNicholas et al. [19] found the death of a pet may cause great distress to owners, especially when the pet has associations with a deceased spouse or former lifestyle and reluctance to part with a pet may lead to non-compliance with health advice. And now raising a pet is also a big economic expenditure. It is necessary to take pets for regular physical examination, washing, nursing and exercise, which is not very friendly to the elderly and people who are not convenient for activities.

In summary, through these studies, we can find that keeping a pet is a good spiritual companion and emotional support for people. However, keeping pets also faces some economic burdens and space constraints. Not everyone is suitable for keeping pets.

6.2 Pet System

Many studies have put forward a variety of pet systems. Within our knowledge, there are a number of researches we could find, according to their different ways of presenting, there are several types of pet systems.

Yuhang Zhao et al. [20] proposed a digital Pet system on a handheld projector. The User can the project pet into the physical environment, and its behaviors and evolution will vary according to the surrounding environment. A plurality of pets projected by a plurality of players can also interact with each other, thereby possibly generating social interaction between players.

Yokoo et al. [21] designed a virtual electronic pet and a pet-type robot that changes emotional state and instinctive state according to surrounding information and internal information. They received and transformed the information of virtual electronic pet between the personal PC and the pet-type robot. So that the action of an electronic pet can be controlled by the internal state of the electronic pet.

Liang Liu and Huadong Ma [22] integrated sensor networks into mobile pet game. They proposed a wireless sensor network-based mobile pet game. Wireless sensor networks enable the virtual pet to perceive the physical world and make their system entertainment and practical.

Cliffen Allen et al. [23] 's recent work simulated pet raising scenarios in users' mobile phones. However, the technology does not yet support real interaction with virtual content.

In summary, there are several kinds of research about the pet system. The device, ways of presenting, user cases have many differences but also share some commons. The commons mainly reflected in the interaction between the pet and the pet owner. They all want to simulate the interaction between real pets and the environment and let the system more smoothly.

6.3 AR Navigation

AR navigation becomes more and more popular. In general, academic research on AR navigation can be divided into these categories.

6.3.1 Marker-Based Tracking

Daniel Wagner and Dieter Schmalstieg [24] described a first stand-alone Augmented Reality system with self-tracking. They achieved their system by using a marker-based tracking toolkit which is ARToolKit. Ryotaro Kuriya et al. [25] proposed a new Augmented reality system that is applied to robot navigation in their research. They used pattern IDs as a marker to control the robot.

6.3.2 GPS Based Tracking

GPS means Global Positioning System. It is a satellite-based radio navigation system owned by the United States government and operated by the United States Space Force [26]. Wolfgang Narzt et al. [27]'s paper presents the navigation system for cars. In the AR navigation system, the 3D depiction of the street is generated from the current GPS position and orientation, the maps, the topography information, and the calculated route technologies, e.g. GPS is the core of their system and is used to keep tracking the car. It helps to locate the current position of the car.

6.3.3 Vision-Based Indoor Positioning (VBIP) Tracking

Hung-Ya Tsai et al. [28] gave a Vision-Based Indoor Positioning (VBIP) AR navigation system. They proposed an algorithm called VBIP which uses visual information and magnetic field to achieve navigation. This algorithm can be divided into two parts, the first part is constructing an indoor map, and the second part is using the preconstructed map. They captured visual features and collected the magnetic field for the AR navigation.

To be a summary, the predecessors in academia have given us a lot of ideas about AR navigation. In our research, we use waypoints to achieve path organization and navigation. It is a simpler and easier operate method.

Chapter 7

Preliminary Evaluation

In order to prove the performance of our system, the experiment aims to study the fun and practical experience that AR pets bring to users and the accuracy of AR navigation. In order to achieve experimental thinking, we will ask participants to use our system. After that, they are required to answer some questions and have a brief interview with us.

The experimental environment is based on the system we have developed.

7.1 Participants

Fifteen participants (6 men, 9 women) aged 21-50 years ($M = 26.80$, $SD = 9.08$) participated in the study. We asked them to rate the AR experience, ranging from 1 (never used AR technology) to 7 (very often used AR technology). Our sample consists of users who have some experience in AR technology ($M = 5.13$, $SD = 2.32$).

7.2 Method

Before starting all participants are given a brief introduction to our system. They then use our system. Each participant was allowed to freely interact with his AR pet and use his AR pet as a guider to experience AR navigation, after which, we asked the participant to answer a set of questions on the experience of interacting with the object.

Some of these questions require participants to evaluate their experience of interacting with AR pets on a 7-point scale, including how similar AR pets are to real pets in shape, sound, and movement. Another problem is to require participants to evaluate the practicality and accuracy of AR navigation in the system. A value of 1 indicates that the participant is very dissatisfied; a value of 7 indicates that the participant is very satisfied. Figure 7.1 shows the questionnaire we used.

After each session, we also asked participants for their suggestions for improving our system.

QUESTIONNAIRE

Name: Age: Gender: Date:

Questions:
These questions are based on 7-point scale **(1: strongly disagree, 7: strongly agree)**.
Answer the questions by circling the most appropriate answer.

1. I am familiar with Augmented Reality system.
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

2. This system is easy to operate.
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

3. This system gives me a feeling of raise a physical pet.
Shape:
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

Sound:
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

Action:
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

4. **The interaction with the AR pet is interesting.**
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

5. The AR navigation part is useful, I can use it find my way easily.
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

6. AR pet as the guider makes the navigation easier and interesting.
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

7. I would like to customize an AR pet for myself.
Strongly disagree Disagree Slightly disagree Neutral Slightly agree Agree Strongly agree

How could the system be improved?

Fig. 7.1 Questionnaire.

7.3 Result

We collected 105 responses (7 questions × 15 participants) from all participants. Table 7.1 shows the result of investigative questions after using our system. And we analysis these data according to the angle of questioning.

Table 7.1 Result of investigative questions after using the system.

Question	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Q1. I am familiar with Augmented Reality system.	2	2	-	-	-	6	5
Q2. This system is easy to operate.	-	-	-	-	-	12	3
Q3.1 This system gives me a feeling of raise a physical pet.-Shape	-	-	-	-	3	9	3
Q3.2 This system gives me a feeling of raise a physical pet.-Sound	-	-	-	-	1	9	5
Q3.3 This system gives me a feeling of raise a physical pet.-Action	-	-	-	1	5	8	1
Q4. The interaction with the AR pet is interesting.	-	-	-	-	1	5	9
Q5. The AR navigation part is useful, I can use it find my way easily.	-	-	-	2	2	10	1
Q6. AR pet as the guider makes the navigation easier and interesting.	-	-	-	-	4	5	7
Q7. I would like to customize an AR pet for myself.	-	-	-	-	-	7	8

Participants involved in this experiment agreed that our system is easy to operate (M=6.20, SD=0.41). As Figure 7.2 shows, in evaluating the similarity between our system and real pets, participants thought that our system gave them a feeling of keeping real pets (M=5.96, SD=0.71). And in the three aspects of shape, sound and action, participants felt similarity sound (M=6.27, SD=0.59) is better than shape (M=6, SD=0.65) than action (M=5. 60, SD=0.73).

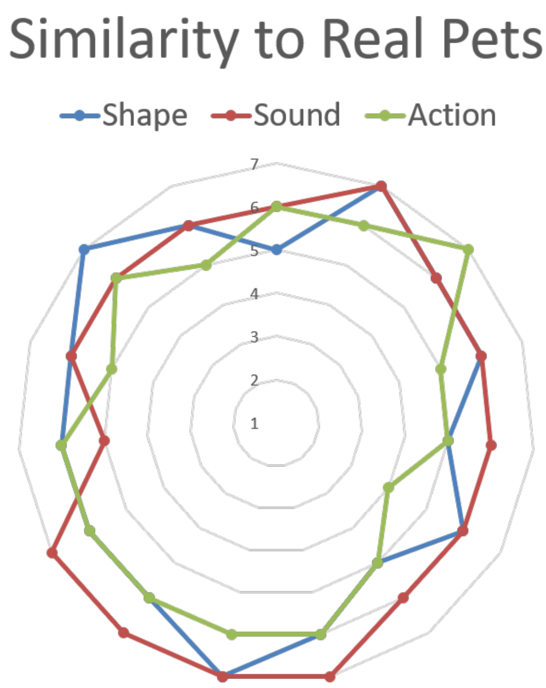


Fig. 7.2 Similarity comparison between system to real objects.

The feedback of AR pet navigation is shown in Figure 7.3. We asked participants to evaluate the performance of AR navigation and judge the impact of AR pets in the navigation section respectively.

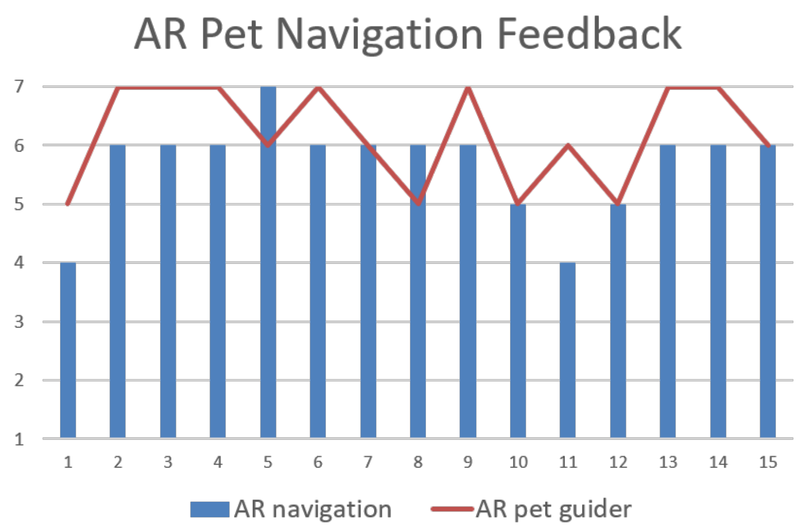


Fig. 7.3 Feedback of AR navigation.

Participants thought that use our system to find way is easily and useful (M=5.66,SD=0.81). And the AR pet as the guider makes the navigation easier and more interesting (M=6.20,SD=0.86). We can find that the AR pet play a positive role in the navigation section.

Most participants agree that the interaction with the AR pet is interesting (M=6.53, SD=0.64). They also expect that our system can have more interactions and make the interaction more realistic. For example, pets can grow slowly from the beginning. If they do not feed for a long time, they will be hungry. If they do not take a bath, the cleanliness will be reduced. In that way, the connection between pets and users would be strengthened.

We also got some other suggestions from the participants. They think that our system can add some voice assistance, which will be more convenient for the elderly. Because the old man will be dazzled and confused after watching it for a long time, and his eyes are not comfortable. In addition, some participants believe that the user interface should be simplified. For example, the source code information in the user authentication interface can be hidden.

Overall, our system get positive feedback from participants. They like this system and are willing to create an AR pet by themselves. One participant said "This system is great for people like me who like pets but hate pet hair." And another participant thought this system made her felt relaxed.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

In this thesis, we proposed an AR pet system that focuses on giving the senior person company and emotional and life support.

Augmented reality is a very popular topic in recent years. The pet system with augmented reality can not only meet the pet raising needs of people who cannot keep pets but also provide a practical and vivid interaction between people and pets. The current AR pet system either pays too much attention to gamification and lacks practical effect, or the interactive form is too rigid and not natural enough. Our target system should satisfy there 3 requirements: the AR pet could be created by the user; the interaction between the pet and the user could be triggered automatically; the AR pet could navigate the user home.

We use a series of photos which uploaded from the user to generate a 3D pet model and put it in the AR scene. In our system, once the user authentication successfully, he can interact with his AR pet. For example, the user can control the pet's position, size and rotation. He also can interact with the pet. The AR pet will trigger different actions according to different physical environments. The navigation part can help the user who wants to go out and are afraid of not remembering the way to find his way home.

Our system designed to reduce the loneliness of people, especially the elderly. Meeting their needs for emotional communication also encourages them to interact more with the outside world. The AR pet can be an object for their emotional sustenance, as well as a channel and tool for their interaction with the outside world.

In order to verify the effectiveness of our system in emotional support and navigation, we invited participants to our evaluation. The feedback is positive.

8.2 Future Work

Our system is a new attempt on how to design a real-world AR pet system. So, our research is mainly to verify the feasibility of the method and technology. The design of application scenarios and the design of specific interactions are a bit rough. In the following research, we can strengthen the work of system emotional support. For example, by adding emotion recognition to users, pets can make corresponding reactions according to different emotions of users, so that they can better accompany users. And as the participants suggested in the questionnaire, we can add some voice prompts to the system to better meet the needs of the elderly.

References

[1] Mark Billinghurst, Adrian Clark, and Gun Lee. A survey of augmented reality. 2015.

[2] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.

[3] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. Recent advances in augmented reality. *IEEE computer graphics and applications*, 21(6):34–47, 2001.

[4] Amanda Edwards-Stewart, Tim Hoyt, and Greg Reger. Classifying different types of augmented reality technology. *Annual Review of CyberTherapy and Telemedicine*, 14:199–202, 01 2016.

[5] Sonia Schechter. What is markerless augmented reality? https://www.marxentlabs.com/what-is-markerless-augmented-reality-dead-reckoning, 2019.

[6] Barbara Resnick. Person-centered approaches to human–animal interaction. *Caring for the Ages*, 20(6):12–20, 2019.

[7] Yamato Kuwahara, Hung-ya Tsai, Yuya Ieiri, and Reiko Hishiyama. Evaluation of a campus navigation application using an ar character guide. In *International Conference on Collaboration and Technology*, pages 242–250. Springer, 2019.

[8] Inc TranSafety. Study compares older and younger pedestrian walking speeds. *Road Management & Engineering Journal*, 1997.

[9] Arthur Juliani, Vincent-Pierre Berges, Esh Vckay, Yuan Gao, Hunter Henry, Marwan Mattar, and Danny Lange. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627*, 2018.

[10] Sigal Zilcha-Mano, Mario Mikulincer, and Phillip R Shaver. An attachment perspective on human–pet relationships: Conceptualization and assessment of pet attachment orientations. *Journal of Research in Personality*, 45(4):345–357, 2011.

[11] Allen R McConnell, Christina M Brown, Tonya M Shoda, Laura E Stayton, and Colleen E Martin. Friends with benefits: on the positive consequences of pet ownership. *Journal of personality and social psychology*, 101(6):1239, 2011.

[12] Marie-José Enders-Slegers. The meaning of companion animals: Qualitative analysis of the life histories of elderly cat and dog owners. *Companion animals and us: Exploring the relationships between people and pets*, pages 237–256, 2000.

[13] Francesca Moretti, Diana De Ronchi, Virginia Bernabei, Lucia Marchetti, Barbara Ferrari, Claudia Forlani, Francesca Negretti, Cleta Sacchetti, and Anna Rita Atti. Pet therapy in elderly patients with mental illness. *Psychogeriatrics*, 11(2):125–129, 2011.

[14] Karen Allen, Jim Blascovich, and Wendy B Mendes. Cardiovascular reactivity and the presence of pets, friends, and spouses: The truth about cats and dogs. *Psychosomatic medicine*, 64(5):727–739, 2002.

[15] Roswitha Scheibeck, Martin Pallauf, C Stellwag, and B Seeberger. Elderly people in many respects benefit from interaction with dogs. *European journal of medical research*, 16(12):557–563, 2011.

[16] Alan M Beck and Aaron H Katcher. Future directions in human-animal bond research. *American Behavioral Scientist*, 47(1):79–93, 2003.

[17] Marguerite O'Haire. Companion animals and human health: Benefits, challenges, and the road ahead. *Journal of Veterinary Behavior*, 5(5):226–234, 2010.

[18] Cecily Maller, Mardie Townsend, Anita Pryor, Peter Brown, and Lawrence St Leger. Healthy nature healthy people:'contact with nature'as an upstream health promotion intervention for populations. *Health promotion international*, 21(1):45–54, 2006.

[19] June McNicholas, Andrew Gilbey, Ann Rennie, Sam Ahmedzai, Jo-Ann Dono, and Elizabeth Ormerod. Pet ownership and human health: a brief review of evidence and issues. *Bmj*, 331(7527):1252–1254, 2005.

[20] Yuhang Zhao, Chao Xue, Xiang Cao, and Yuanchun Shi. Picopet: " real world" digital pet on a handheld projector. In *Proceedings of the 24th annual ACM symposium adjunct on User interface software and technology*, pages 1–2, 2011.

[21] Naohiro Yokoo, Yasuhiko Kato, Masakazu Hattori, Masahiro Fujita, and Naoyasu Hosonuma. Electronic pet system, network system, robot, and storage medium, August 8 2006. US Patent 7,089,083.

[22] Liang Liu and Huadong Ma. Wireless sensor network based mobile pet game. In *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games*, pages 30–es, 2006.

[23] Cliffen Allen, Jeanny Pragantha, and Darius Andana Haris. 3d virtual pet game" moar" with augmented reality to simulate pet raising scenario on mobile device. In *2014 International Conference on Advanced Computer Science and Information System*, pages 414–419. IEEE, 2014.

[24] Daniel Wagner and Dieter Schmalstieg. First steps towards handheld augmented reality. In *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings.*, pages 127–135. IEEE, 2003.

[25] Ryotaro Kuriya, Takeshi Tsujimura, and Kiyotaka Izumi. Augmented reality robot navigation using infrared marker. In *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 450–455. IEEE, 2015.

[26] US DoD. Global positioning system standard positioning service performance standard. *Assistant secretary of defense for command, control, communications, and intelligence*, 2001.

[27] Wolfgang Narzt, Gustav Pomberger, Alois Ferscha, Dieter Kolb, Reiner Müller, Jan Wieghardt, Horst Hörtner, and Christopher Lindinger. Augmented reality navigation systems. *Universal Access in the Information Society*, 4(3):177–187, 2006.

[28] Hung-Ya Tsai, Yamato Kuwahara, Yuya leiri, and Reiko Hishiyama. *Vision-Based Indoor Positioning (VBIP) - An Indoor AR Navigation System with a Virtual Tour Guide*, pages 96–109. 08 2019.