

Input: User speech input, camera image, preferred language

Output: Generated and recommended 3D objects in the MR environment with generated voice messages

Initialize MR Environment:

Play welcome message via headset audio;

Prompt user to select preferred language;

Step 1: Language Translation and Speech Processing

speech_input $\leftarrow$ record_user_speech();

if *selected_language* == "English" **then**

 transcribed_text $\leftarrow$

SeamlessM4T.transcribe(speech_input); **end**

else

 translated_text $\leftarrow$ SeamlessM4T.translate(speech_input, selected_language);

end

Step 2: Object and Attribute Extraction from Text

optimized_llm $\leftarrow$ load_Llama2_7B_local(backend = llama.cpp);

prompt $\leftarrow$ create_prompt(translated_text, using = LangChain);

object_data $\leftarrow$ optimized_llm.extract_objects(prompt);

Step 3: Contextual Object Recommendation

recommended_objects $\leftarrow$ optimized_llm.recommend_objects(object_data);

related_objects $\leftarrow$ search_repository(object_data);

play_audio_description(recommended_objects);

Step 4: Visual Scene Augmentation

if *user_triggers_camera* == *True* **then**

 scene_image $\leftarrow$ capture_camera_image();

 scene_objects $\leftarrow$ VLM.detect_objects(scene_image);

 visual_recommendations $\leftarrow$ optimized_llm.recommend_missing_

objects(scene_objects); recommended_objects.append(visual_

recommendations);

 play_audio_scene_description(scene_objects, visual_recommendations);

end

Step 5: Real-time 3D Object Generation

foreach *obj* **in** *recommended_objects* **do**

 model3D $\leftarrow$ generate_3D_model(obj);

 simplified_model $\leftarrow$ simplify_

mesh(model3D); render_in_

MR(simplified_model);

 save_to_repository(simplified_model, obj);

end

Algorithm 2: Matrix MR Application for Context-Aware 3D Object Generation.