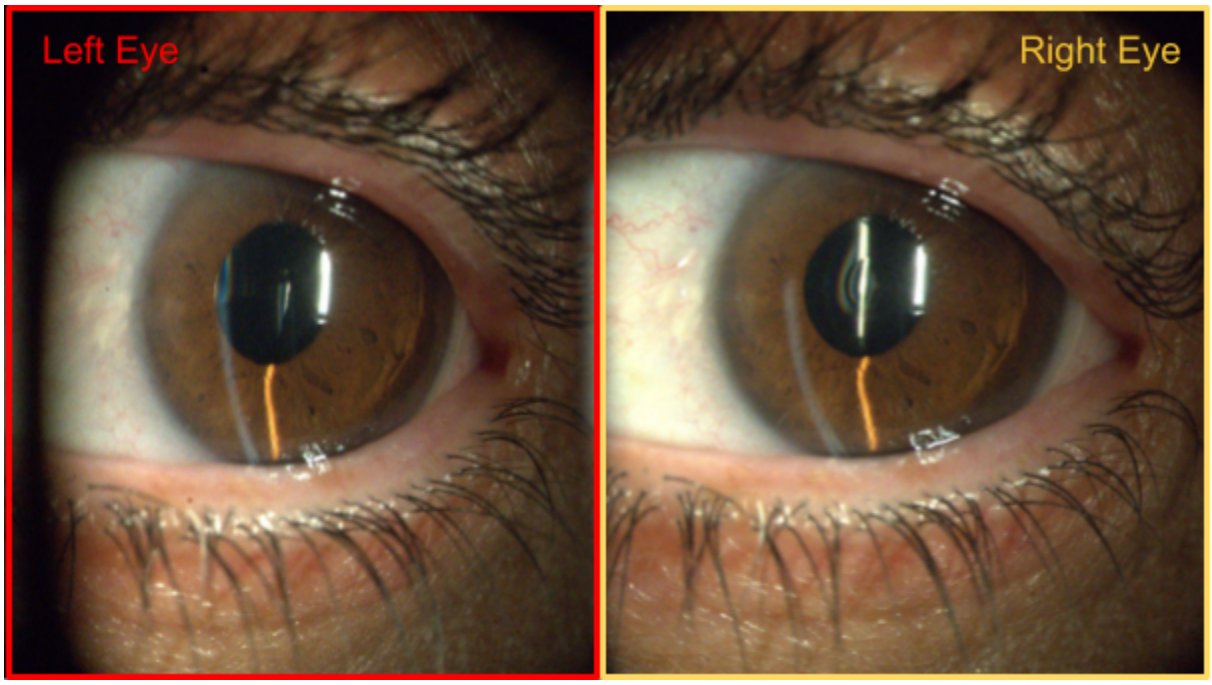
Unity Version: 2000.6.26f1

Eyes (Background)

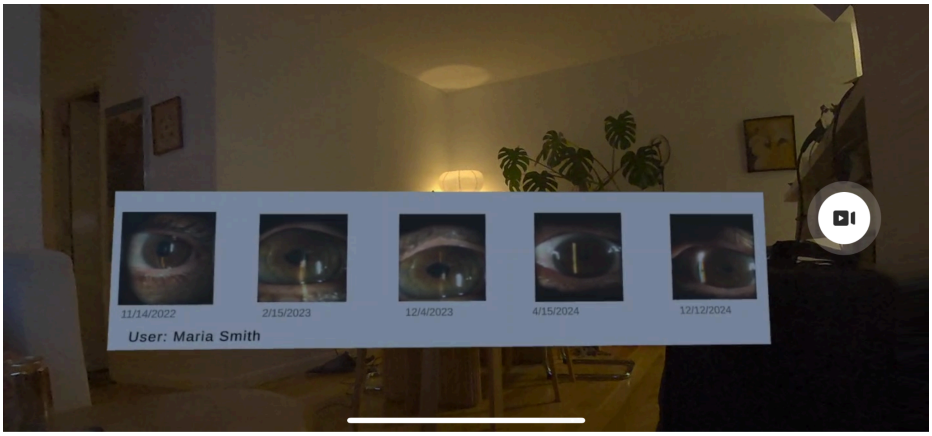


The examining the eye app utilizes stereoscopic rendering on the Meta Quest 3. Images from a split microscope of the eye allows an image to be viewed in stereo and appears to be 3 dimensional. This is achieved by rendering the left-skewed image in the left eye and the right-skewed image in the right one. This generates a stereoscopic effect, for the purpose of medical doctors and students an enhanced ability to view microscopic eye images without a stereoscope.

UI Components:

When you open the app, you will see the following main components:

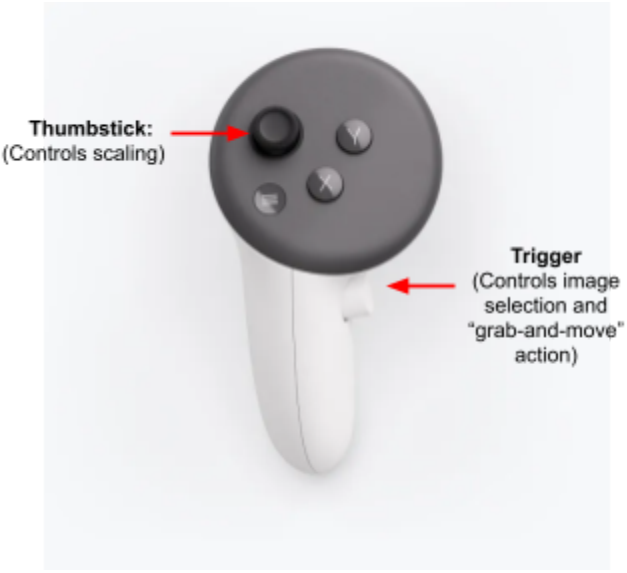
1. **Image display:**
    a. This is the central focus of the app, displaying a large rendering of the “default” eye image.
    b. **User Interaction:** By using controller interaction “move” and “scaling” (see below for controllers), you can 1. Move the image (movement), 2. Adjust the size of the image (scaling).
2. **Thumbnail selection panel (library)**

a. 

b. This is located at the bottom of the UI, allowing users to view a patient's eye images, to provide a gallery of thumbnails for quick image selection.
c. **User interaction:** Users can interact with the thumbnail selection panel or can directly interact with the panel itself, moving it around the space.
    i. Use the library to change image display (1): use the controller to point at the thumbnail you want to see on display, and press down the index trigger. This selects a thumbnail and will display the corresponding one.
        1. Note: Users also have the ability to directly "tap" on the images on the panel for selection. Similar to pushing a button, users can simply use their hands to press down on the thumbnail they want to see. This action must extend until the image is past the white panel), so this is a large pressing motion.
    ii. Move the library itself: Similar to image display, you can use the controllers to "select" the panel itself. This will initiate the "grab" action, and you can move around the panel in space. This requires you to aim the controller at the panel, so the white portion of the panel.

3. **Viewer pane**
    a. This is a group of three panels together, and the pane consists of three side-by-side image displays for simultaneous image comparison. This is above the large image display.
    b. The panels display that you are able to view multiple images at the same time in stereo.
    c. User interaction: Same as interaction for "image display".

Controllers:
The Meta Quest controllers are he essential tools for interacting with the app. The controllers are used for 1. **selection** to either select an image on the image pane or to move an image, or 2. **scaling** and image.



Primary controller actions:

1. **Selecting** images:
    a. Point the controller at the thumbnail in the image selection pane, seen using the controller ray.
    b. Press the index trigger button to select the thumbnail.
2. **Scaling** an image:
    a. Point the controller at either the main image in the image display or one of the three images in the viewer pane.
    b. When the controller is pointed at the image, use the thumbstick (up/down) to scale the image size.
        i. Push up for enlargement.
        ii. Push down to shrink/minimize.
3. **Move** an image:
    a. Point the controller at the image, using the controller rays.
    b. Hold the grip button (trigger), and *while* holding down the trigger, move the controller to reposition the image in 3D space.
    c. Note: This can be used to translate and rotate an image across all three x, y, z axes.

What the user sees:

- **Unity Background**: This app is in pass-through mode, meaning you can see your surroundings. This is to allow the medical doctors to interact and be aware of their environment while looking at the images.
- **Stereoscopic Effect**: Images render differently in each eye for a 3D viewing experience.

User flow summary:

1. After you open the app, see the default image in the **image display** panel.
2. Look above the default image to view the **viewer pane** of three side-by-side images.
3. Access the **thumbnail selection panel** to change the image in the image display.
4. Use the controllers to “select” a thumbnail using the trigger button, this is displayed on the image display panel.
5. Use the controllers to scale, move, and switch between images.
6. Move elements within the 3D space to users discretion to best interact with elements.

Packages:

1. **XR Plugin Management:** Enables XR development
    a. Go to **Edit > Project Settings > XR Plugin Management**.
    b. Install and enable the **Oculus** plugin (or **OpenXR** for cross-platform support).
2. **Oculus integration assets:** Enables XR development for oculus assets
3. **TextMeshPro:** To get texts like labels into the pane
    a. **Purpose**: Provides advanced text rendering for search fields, labels.
    b. **Where to Find**: Unity Package Manager.
    c. **Configuration**: Automatically installs when you create a new Unity project if selected.
4. **XR Interaction Toolkit:** Gets prefabs and tools for VR interactions
    a. **Where to Find**: Unity Package Manager.
    b. **Configuration**:
        i. Install the package and enable: **Edit > Project Settings > XR Plugin Management**.
5. Meta XR All-in-One Display:
    a. Where to find: Unity Package manager.
    b. Configuration:
        i. Install the package and enable: **Edit > Project Settings >** Meta XR All-in-One Display.
    c. Add **OVRCameraRig** (if using Oculus Integration) or use **XR Origin (VR)** (if using XR Interaction Toolkit).

Settings:

1. Input system: Supports advanced input configurations (controller, joysticks, triggers)
    a. **Where to Find**: Unity Package Manager.
    b. Enable the **New Input System** in **Edit > Project Settings > Player > Active Input Handling**.
2. **Universal Render Pipeline (URP):** Allows enhanced graphics performance, good setting.
    a. Where to find: Upon configuring the project.
3. Ensure that **Stereo Rendering Mode** is set to **Single Pass Instanced**
    a. **Project Settings > XR Plug-in Management > Oculus**.
4. Ensure that Android is selected in build settings. Deploy to device.