1 Application Description

I implement the game "Towers of Hanoi" as an AR mobile game (Android), while using the Vuforia Library. As my towers I used 3 Bottles, which are brought into the application as Vuforia cylinder targets. For these cylinder targets I also added occlusion objects, which can be done in the Unity Inspector, when selecting a cylinder target. These occlusion objects handle the occlusion between real and virtual object.

As game stones I use Tori (Torus), which are created at runtime from a Cube Unity Object. To achieve this I used the code from the *Primitives* script posted by *Steffen* on the Unity Forum (link to the forum thread) and created a Torus prefab, which is then instantiated in the tower handler. These Tori are then attached to one Tower and can then be moved to other towers.

1.1 Game Stone Interaction/Movement

1.1.1 Game Stone Selection

There are 2 ways to select a game stone.

- Selecting the top most game stone from a tower by using the virtual button
- Selecting the top most game stone by touching it (raycasting with game stone)

When a game stone is selected it moves above the tower and changes its color to red - i.e. red means it is selected.

If a selected game stone is touched again it is deselected and returns to its original position.

1.1.2 Game Stone Placement

There are 2 ways to place a game stone.

- Placing the game stone onto a specific tower by pressing the virtual button in front of the specific tower
- Placing the game stone onto a specific tower by touching the tower (raycasting with tower)

1.1.3 Raycasting

For the raycasting I created an empty GameObject to which the *RayIntersectionHandler* is attached. In the *Update*-function of the *RayIntersectionHandler* I check if the screen of the mobile device was touched and if the touch hit either a game stone or a tower (this is a great resource for handling the touch and this resource helped me to implement the raycasting). If a game stone is touched, it is selected if it is a playable stone (only the top stone can be selected) or de-selected if it is currently selected. If a tower is touched, while a game stone is currently selected the game stone will be placed onto the touched tower. When a tower is touched, while no game stone is selected nothing happens.
As the towers are cylinder targets, which cannot be hit by the raycast I placed cylinder objects inside of the cylinder targets, which can then be hit by the raycast (these cylinders can be spotted in the video, when looking inside the bottles from the top).

1.1.4 Virtual Buttons

For the virtual buttons I created a script (*VirtualButtonHandler*), which handles the button presses. This script is attached to the image target the virtual button is placed on and holds references to this virtual button and to the tower the virtual button belongs to. By pressing the virtual button one can either select the top-most game stone from the tower or place the currently selected game stone onto this tower.

In the video all the aforementioned functions are showcased.

2 Chosen Targets

As we have three towers we need 3 cylinder targets and 3 image targets for the virtual buttons. I tested all the chosen images in the Vuforia target manager and as they all got 5 stars I used them.

Important to note is, that the amount of maximum simultaneously tracked targets has to be set to 6 in the Vuforia configuration in the Unity inspector.