

Vignette

Veerle Vjverberg

2023-05-26

Title: I-Witness: Video Observation and Recall Shiny App

Introduction

The I-Witness Shiny App is a powerful tool designed to assess video observation and recall skills. It presents users with a video clip depicting a crime, followed by a series of interactive tasks aimed at evaluating their ability to accurately describe a perpetrator, make lineup decisions, and rate their confidence in their decisions. Before completing these tasks, there is a distractor task wherein they guess the country that various flags belong to.

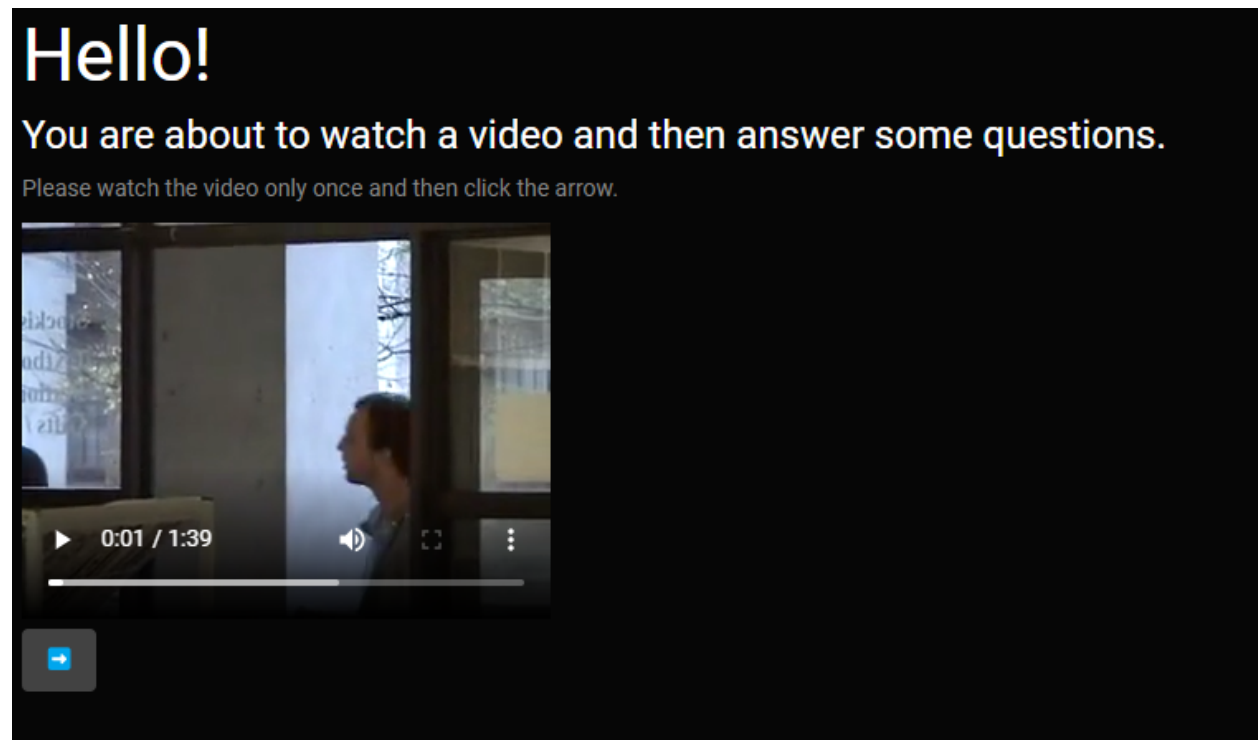
This vignette provides a detailed overview of the app's functionality and usage.

App Workflow

Video Observation:

The app begins by displaying a welcome message and instructions to watch the video. Users will watch one of two videos, depicting different perpetrators steal a book from a bookshop. They are invited to play the video and must watch it carefully. After watching the video, they can proceed by clicking the arrow button.

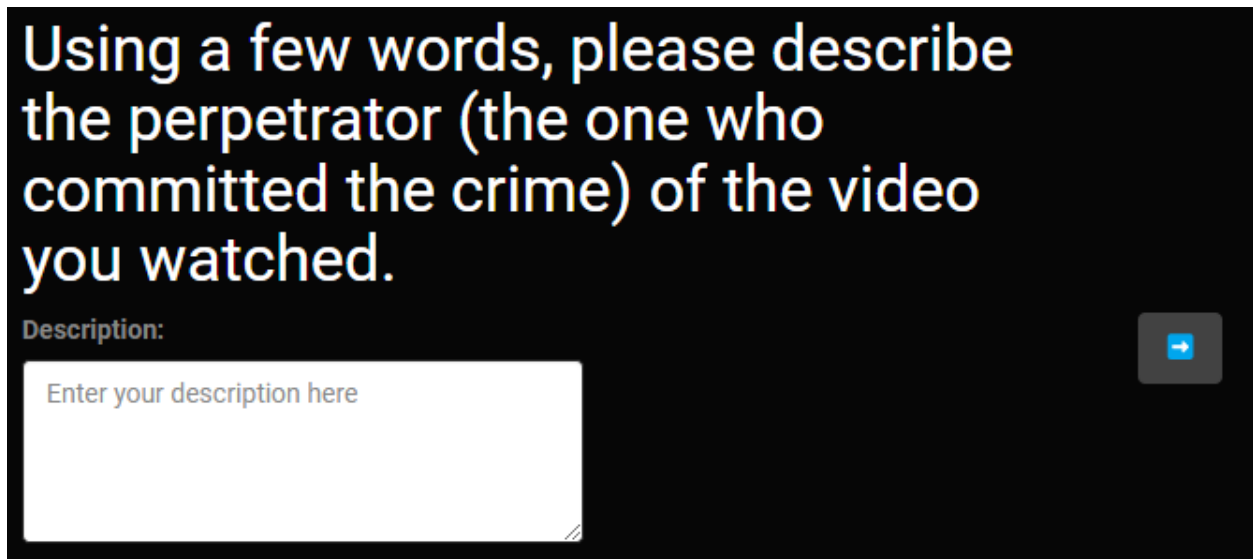
```
knitr::include_graphics("VignetteImages/Video.png")
```



Perpetrator Description:

Upon clicking the arrow button, users are prompted to describe the perpetrator of the video they watched. A text area is provided where users can enter their description in a few words. Once they have entered their description, they can click the arrow button to continue.

```
knitr::include_graphics("VignetteImages/PerpetratorDescription.png")
```

The screenshot shows a dark-themed user interface. At the top, large white text reads: "Using a few words, please describe the perpetrator (the one who committed the crime) of the video you watched." Below this, the word "Description:" is displayed in a smaller white font. To the left of "Description:" is a white rectangular text input field with the placeholder text "Enter your description here" in a light blue font. To the right of the input field is a small, dark gray square button with a white right-pointing arrow inside.

Distractor Task:

Next, users are presented with a series of flag images. Each flag corresponds to a country, and users are required to name the country for each flag. Text input fields are provided for users to enter their answers. Once they have completed the task, they can click the arrow button to proceed.

```
knitr::include_graphics("VignetteImages/DistractorTask.png")
```

Name the country of these flags below

 <p>Flag 1</p> <input type="text"/>	 <p>Flag 2</p> <input type="text"/>	 <p>Flag 3</p> <input type="text"/>
 <p>Flag 4</p> <input type="text"/>	 <p>Flag 5</p> <input type="text"/>	 <p>Flag 6</p> <input type="text"/>



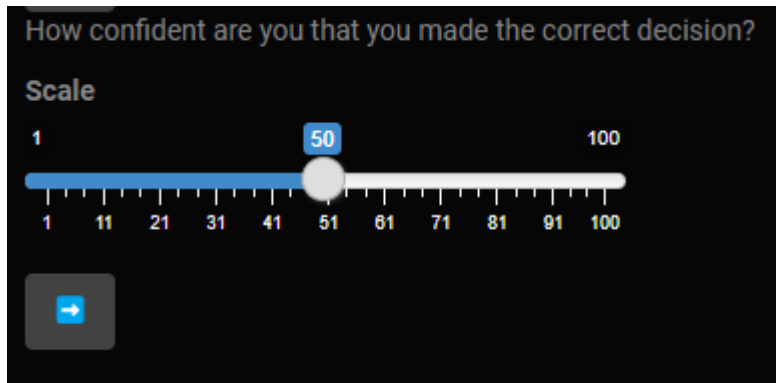
Lineup Decision:

Users are then shown a lineup image containing several individuals. They are informed that they need to identify the person who stole the bag based on their video observation. A text input field is provided for users to enter the number corresponding to their lineup decision. If they believe the person is not present, they can enter 0. After making their decision, they can click the arrow button to continue.

Confidence Rating:

Users are asked to rate their confidence in the lineup decision they made. A slider input is provided, allowing users to select a value between 1 and 100 to represent their confidence level. After selecting their confidence rating, they can click the arrow button to proceed.

```
knitr::include_graphics("VignetteImages/Confidence.png")
```



Feedback:

Finally, users reach the feedback stage, where feedback specific to their performance is displayed.

Implementation

The I-Witness Shiny App is implemented using the Shiny framework in R. It makes use of various Shiny components and functions to create an interactive user interface and handle user interactions.

The ui function defines the layout and structure of the app's user interface. It consists of different conditional panels that are displayed based on the user's progress through the tasks. The panels include video playback, text entry for perpetrator description, flag identification task, lineup image display, confidence rating slider, and feedback display.

The server function handles the app's logic and reactive behavior. It sets up resource paths for video and flag images, randomly selects a video from the available options, updates UI elements based on user interactions, and handles events triggered by button clicks or inputs.

Usage

To run the I-Witness Shiny App, execute the runApp function. The app will launch in a web browser and present the user interface described above. Users can navigate through the tasks by following the instructions and interacting with the provided elements.

Conclusion

The I-Witness Shiny App offers an engaging and interactive platform for assessing facial recognition and recall skills. It can be utilized in various research or educational settings where accurate description and identification of observed events take the forefront. Researchers, educators, or individuals interested in studying eyewitness identification can leverage this app as a valuable tool.