

A Virtual Microscope for Use in Online and Onsite Biology Labs

T. Kawakami¹, G.M. Cohen², and J. Zhong¹

¹ Department of Computer Science, Troy University, Troy, AL 36082

² Department of Biological and Environmental Sciences, Troy University, Troy, AL 36082

Distance learning has expanded and redefined the educational environment by offering access to almost every discipline independently of time and location. However, conceptual subjects are more successfully presented than those requiring equipment and/or manual skills. For this reason, virtual microscopes must provide equivalent functions to simulate their laboratory counterparts [1]. In the present study, we improved and extended the features of the virtual microscope so that it closely corresponds to the functions of student microscopes [2].

Since the previous work [2], we have improved the virtual microscope by adding three new features. First, we upgraded the library for presenting the 3D view of the virtual microscope by replacing the older O3D JavaScript library used in the earlier study [2] with another JavaScript library, Three.js [3], for two reasons: 1) O3D is now obsolete and 2) Numerous developers adopted Three.js and several textbooks refer to it. In addition, active community sites formed for discussions, issuance of reports, and support for developers to initiate web 3D programming with Three.js. The 3D object files created on Blender are imported to applications using Three.js. To display and manipulate the microscope, an html file was created to import both the 3D object files and JavaScript code that handles its manipulation. With the 3D files from Blender and JavaScript codes using Three.js, the browser displays the microscope that the users control. The view through the eyepieces is displayed in a 2D format. The image is shown to the web page with the JavaScript code working with HTML5 that will be the recent markup language to construct the structure of the page and added new features such as <canvas> element to support for user's interactivity and multimedia content.

Second, we added a focusing function that employs a blurring algorithm on the downloaded image to simulate focusing. We employed Pixastic.js [4], a JavaScript library for image processing in which one of its functionalities is to configure the intensity of the effect and apply it by collaborating with HTML5. The blurring effect is shown in Figure 1. We chose the blurring algorithm because it responds quickly and also eliminates the need for downloading and linking stacks of overfocused and underfocused images to simulate focusing.

Third, we added labels with pointers to identify specific structural details of the specimen, along with a short explanation of their cellular functions as shown in Figure 2. In the example shown, the explanations on the right-hand side remain while pointers appear and disappear based on the user's preferences. However, many other configurations are possible and can be changed depending on the microscope slide and assignment. As a result, both students and teachers can use the pointers and explanations for assignments and testing. For operational ease, we constructed a transparent image file that contains these labels with arrows and drew the image on that of the specimen by using HTML5 canvas element.

In the present study, we updated the background technology for controlling 3D rotation to provide users with interactive use of the virtual microscope and also added features for focusing (blurring and

deblurring) and labeling images on the microscope slides. The labeling feature can also be used for assignments and testing. The virtual microscope also serves a pedagogical value for onsite students by preparing them for their regular labs through assignments and exercises. In summary, the virtual microscope described herein closely corresponds to the features and operations of student microscopes.

References

- [1] M.A. Bochicchio and A. Longo, *IEEE Trans. Learning Technologies* 2 (2009), p. 320.
 [2] T. Kawakami *et al*, *Microsc. Microanal.* 18 (Suppl 2), 2012, p. 40.
 [3] Three.js, <http://mrdoob.github.com/three.js>
 [4] Pixastic.js, <http://www.pixastic.com>
 [5] Supported in part by a Troy University Faculty Development Grant.

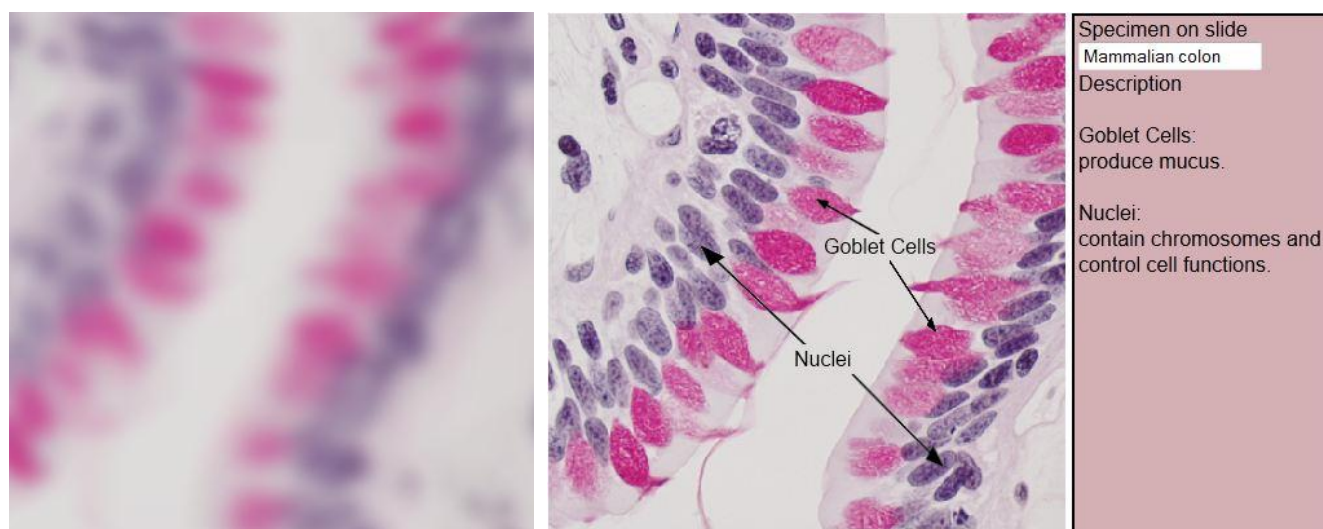


Figure 1. Unfocused (blurred) specimen. **Figure 2.** Focused (deblurred) specimen with labeled cells.