Exploring the Feasibility of Mosquito Count Automation Through ImageJ

Nathaniel Boateng, Ashwin Roperia, Daniela Cabrales, Prayag Sreenivasan, Logan Sandell, Micaela Geborkoff, Foluso Osoba

Abstract

The NASA-SEES 2021 Summer Internship affords high school students the opportunity to conduct research that advances Earth science. The SEES Earth System Explorers-Mosquito Mappers team investigates the relationship between mosquitoes and the environment in the context of human health. As apparent during Mosquito Mapper fieldwork, manually counting mosquitoes in a breeding habitat aids in the understanding of mosquito ecology. Absent a scientific approach, however, manual counts are error-prone and are deemed questionable for use in mosquito management models. To ensure that these counts inform meaningful scientific outcomes, the counting process needs optimization. As such, we explored the feasibility of automating the mosquito count process while minimizing error using ImageJ, an open-source image processor. To this end, images captured during the volumetric sampling of mosquito traps and supplemental images obtained from the GLOBE database were processed using ImageJ. A comparison of the manual and ImageJ counts revealed that both count types were largely unreliable as the difference between many of them exceeded a tolerable margin of error, and no count type was consistently more reliable due to citizen-scientist technique and software limitations. Thus, the results do not support automation using ImageJ. Rather, they indicate that ImageJ's performance depends on the "quality" of the image samples, thereby underscoring the need for standardized scientific methods in the mosquito counting process. However, it is improbable that citizen scientists will employ the counting methodologies of expert scientists since citizen scientists generally prize convenience over validity. An optimal solution may therefore involve a more robust algorithm that builds on the strengths of ImageJ and eliminates the citizen-scientist manual count upon integration into the GLOBE Observer Mosquito Habitat Mapper tool. Although more research is needed to assess the cost-effectiveness, such a multi-layered solution would assist scientists' prediction of mosquito populations and management of mosquito-borne diseases.

Key words: mosquito count, automation, data quality, ImageJ, citizen science

---

Introduction

Deforestation and urbanization have eliminated traditional vector habitats. As a result, vectors have evolved to coexist with humans (Little et al.). Thus, a robust vector surveillance system is needed to mitigate vector-borne diseases. This includes monitoring phenological cycles through traps and periodically collecting count data. Since quantification is laborious, scientists have explored avenues to make the process more efficient and reliable. For example, some have used germination paper, scanners, and ImageJ to quantify mosquito oviposition. Others have conducted analyses on different subsampling methods, one of them ImageJ, to determine the method that best predicts the mosquito quantity and species composition of a larger sample (Jaworski et al.). Recently, to automate the black fly count, scientists trapped, sorted, and photographed flies before analyzing relevant samples in ImageJ (Parker et al.).

Though these counting methodologies may be considered efficient and accurate by professional researchers, citizen scientists, who often prioritize convenience over validity, are less inclined to employ such procedures. Citizen scientists are needed, however, because they extend the spatiotemporal footprint of scientists (McClure et al.). Stated differently, citizen scientists are "boots on the ground" and provide "strength in numbers." Even still, citizen scientists' resource constraints are often overlooked in scientific literature. Considering the importance of their data, although imperfect, we seek to bridge a gap in hopes of eliminating the citizen-scientist manual mosquito count. Hence, based on observation and research, we hypothesized that through robust technology (e.g., machine learning), citizen-scientist images can lead to a more efficient and reliable mosquito count, a count which informs mosquito management models (McClure et al.). Factors that may affect the outcome include the software automation and related process, the citizen scientists' sampling technique, the identification of targeted specimens, the mosquito habitat, image quality, and human bias. Through ImageJ (version 1.53a), we investigate the practicality of automating the citizen-scientist mosquito count while minimizing error and offer relevant recommendations.

Methods and Materials

Mosquito Habitat
As part of the SEES internship, some interns were required to construct homemade mosquito traps and conduct experiments to study mosquito oviposition habits, identify larvae belonging to Aedes, Anopheles, and Culex, and perform source reduction. An intern from our team assembled three mosquito traps in the natural environment. Over a span of five weeks, the traps were subjected to the Texas summer heat, heavy rains, and gardening (Experiment 2 only) and were reset on average every nine days, resulting in four experiments. The mosquito traps resided in an eco-friendly area of deciduous, broad-leaved trees, limestone rocks, fertile soil, and manicured and short wild grass (Figure 1). The container habitats included one six-gallon black bucket and two five-gallon camouflaged buckets each with surface areas of approximately 616 square centimeters and 573 square centimeters, respectively (Figure 1). Inside each trap was one rock and one stick collected from the surroundings (Figure 1). The water occupied 80% of each bucket and was reduced by 40% for the third and fourth experiments. Scattered across the surface of each bucket, dog chow attracted the mosquitoes for the first three iterations and was omitted in the last iteration. At each experiment reset, the water was disposed of and each bucket was cleaned.

Mosquito Trap Sample
Although we successfully lured mosquitos to oviposit in traps, counting and identifying mosquitoes proved difficult. Mosquitoes were represented from all stages from eggs to miniature larvae to adults. The volumetric sampling (Figure 1) technique enabled a rough estimate of the seemingly innumerable mosquitoes or larvae. With volumetric sampling, the citizen scientist, instead of manually counting the numerous larvae in each trap, obtained a representative sample (e.g., 300 mL) from the trap three times. However, it was noticed that most larvae migrated to the bottom upon disturbance of the habitat, potentially affecting the count. The images were captured for 33 of the 36 volumetric samples using either an iPhone 8 or an iPhone X, and counts were recorded. Though this method made sampling more manageable, estimating the larvae remained tedious and compromised the accuracy of the count. The mosquitoes were inspected while conducting the sampling and a small subset from samples were extracted and examined using a clip-on microscope (Figure 1).

Figure 1. Selected images captured from the mosquito habitat site used in this study.

GLOBE Data Sample

Using the “OLD—GLOBE Mosquito Habitat Mapper Metadata” datafile (Figure 2), records with larvae counts of zero and greater were extracted. The data was then narrowed to those with water-source images. This process provided fourteen sample images for analysis in ImageJ.

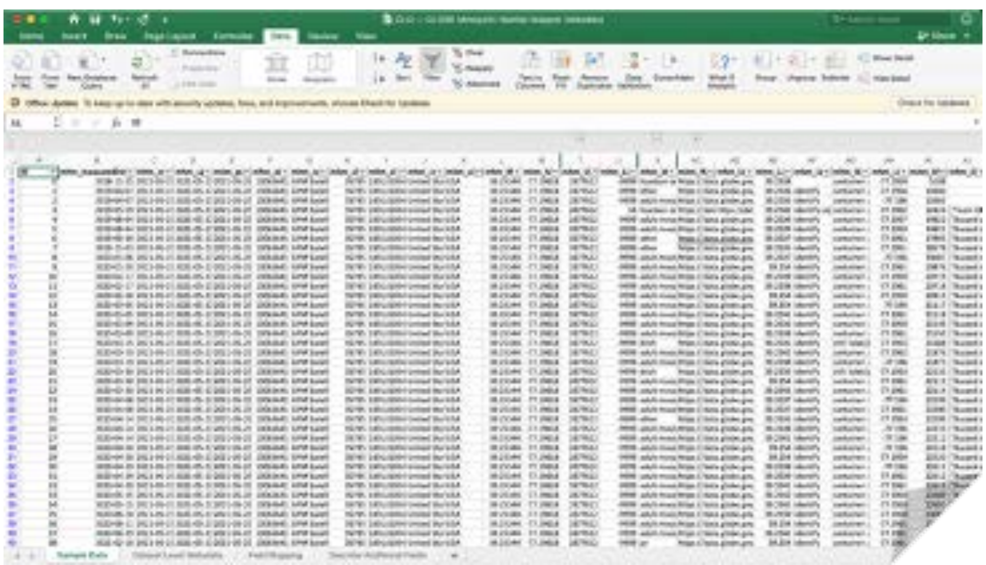

Figure 2. The OLD--GLOBE Mosquito Habitat Mapper Metadata file, representing a population of mosquito habitat data, was used to select GLOBE data samples. (N=14)

ImageJ Processing: Manual Approach

ImageJ is an image processing program used for a variety of purposes. It is used to display, edit, analyze, and process 8-bit, 16-bit, and 32-bit images. Its extensive application in the biological science field (Rueden et al.) made it a logical candidate for the mosquito count automation research project. As such, we downloaded version 1.53a of ImageJ from the ImageJ website and performed the steps outlined in Figure 3. For each of our 47 sample images, ImageJ completed the analysis process within minutes (Figure 4).

Figure 3. The *manual* sequence for processing image samples in ImageJ.

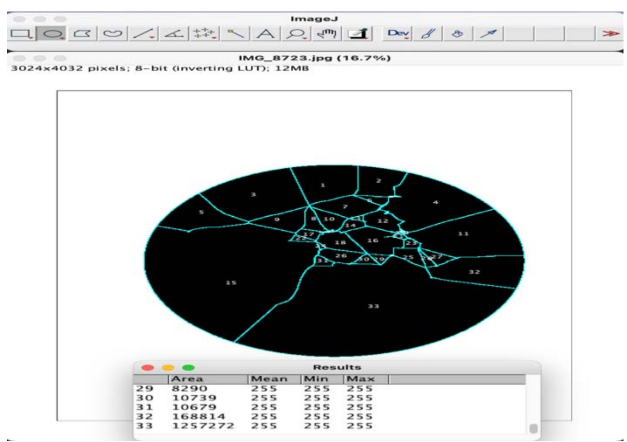

Figure 4. A completed analysis of an image using ImageJ. The software identified 33 larvae, as shown in the CSV file ("Results").

ImageJ Processing: Macro Approach

During the research process, the team observed that manually executing the steps outlined in Figure 3 for each image increased the chance of user error. Consequently, the *AutomatedMosquitoCounter* macro was coded as a more efficient alternative (Figure 5). The macro eliminates several manual steps (Figure 3) and essentially requires the end-user to: (1) input the image, (2) capture the region of interest (ROI), and (3) adjust the threshold (Figure 6). Over time, this reduces processing time and the potential for user error. However, of the remaining manual steps, "Adjust Threshold" is inherently the most subjective and error-prone.

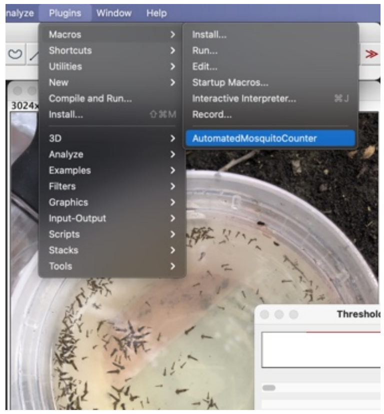

Figure 5. The image shows the installed macro's location in ImageJ. After the user selects the macro, it runs and opens the threshold tab, where the user must adjust the threshold until the specimens are uncovered. Then, the CSV file indicating the mosquito count is produced.

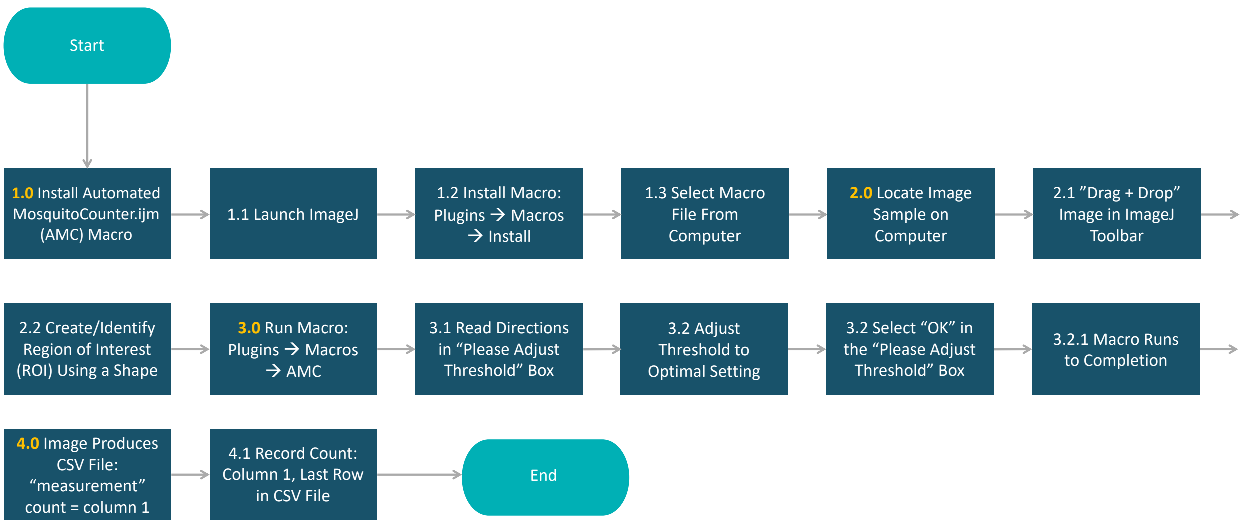

Figure 6. The *macro* sequence for processing image samples in ImageJ.

Results

Mosquito Trap Experiments

There were significant differences in the manual and ImageJ counts across mosquito trap experiments. The manual counts were mere estimates; some were reasonable, most were not (Figure 7). The sheer quantity of larvae, the larvae's dispersal, and a cloudy film that settled on the surface affected the outcome of the count for Experiment 1. The suspended debris in Experiment 2 negatively influenced the count. Image processing resulted in a reliable count for only sample 20 (516 count) in Experiment 3 because the larvae congregated at the surface with minimal overlapping. By contrast, the manual counts for the experiment were undercounted. Experiment 4 boasted the greatest accuracy for manual counts because the quantities were visibly small. The ImageJ count, however, is unreliable because of the dirt present in the container (Figure 7). It was observed that the manual count was most reliable when the number of larvae were small. ImageJ's count was most accurate when the larvae were congregated on the surface without overlapping, when there was no debris or foliage, when the container was free of dark hues, when there were no foreign species, and when image quality was strong.

GLOBE Data

The results from the GLOBE Database sample were consistent with our team's mosquito habitat data. The manual and ImageJ counts vary. Also, the larvae count in GLOBE and the ImageJ count appear unreliable for non-container habitats and habitats that lack contrast. This is reflected in the three spikes in the graph (Figure 8).

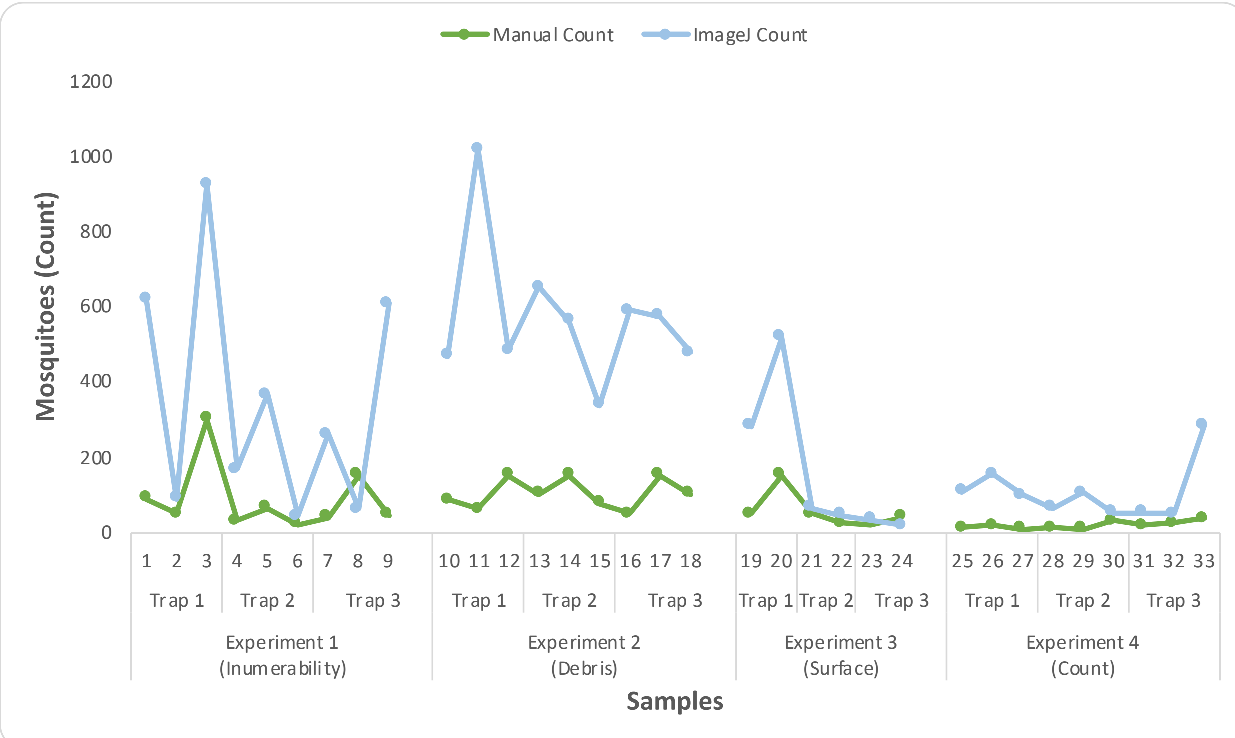

Figure 7. The figure compares manual and ImageJ counts from our mosquito trap data. Each experiment included three traps, where three samples were extracted from each trap, except in Experiment 3. Innumerable larvae, substantial debris, larvae at surface, and dirt impacted counts in the experiments. (N=33)

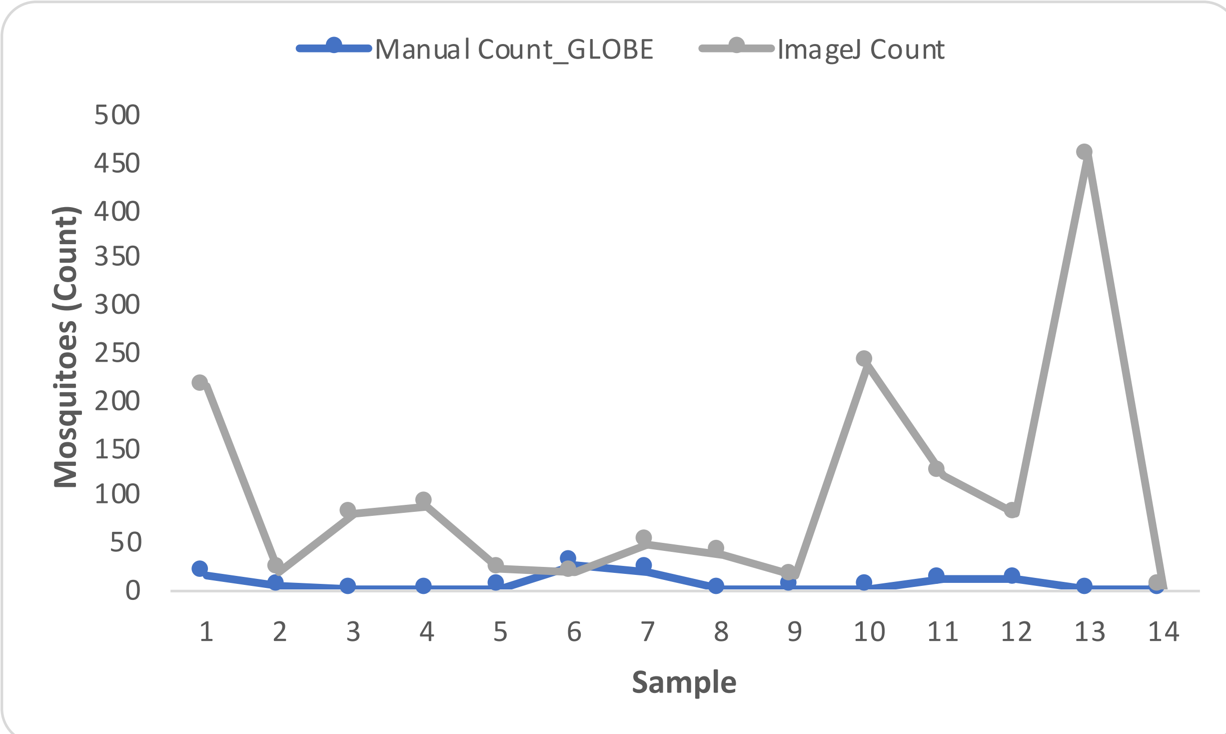

Figure 8. The comparison of manual and ImageJ counts using data from GLOBE. (N=14)

Discussion

While our results partially confirm our hypothesis in that ImageJ is more efficient, the results contradict the notion that ImageJ's, when processing citizen-scientist mosquito habitat images, count is more reliable than the manual count. Recognizing no system is perfect, ImageJ is unreliable in part because of its sensitivity to haphazard techniques (Mains et al.). Not even human intervention (e.g., the adjustment of the image threshold) compensates for this sensitivity. Our research revealed that ImageJ's performance is limited to the quality of the inputted data. In the case of our analysis, certain factors – three-dimensional space, liquid, debris, overlapping larvae, color, mosquito life stage, and foreign organisms – hindered ImageJ from attaining peak performance. Thus, contrary to the 95 percent confidence rate ImageJ yielded when counting black flies (Parker et al.), our results indicated zero percent confidence in ImageJ's ability to process citizen scientists' non-standard scientific approach.

Conclusion

Despite ImageJ unsuccessfully automating citizen-scientist mosquito counts, automating such counts remains possible. One solution devises a methodology that conforms to ImageJ's standards. This includes sampling the container habitat, removing all debris, water, and foreign elements from the sample; spreading the mosquitos onto white printer paper or in a clear, shallow petri-dish; and photographing the sample for analysis in ImageJ. Though this methodology should yield a reliable count, this undermines the efficiency aspect of mosquito count automation since it still relies heavily on citizen-scientist labor and willingness. Another, more ideal, solution adapts ImageJ's strengths to suit citizen scientists' expectations – efficiency and convenience. This alternative solution may comprise a robust algorithm that instantly removes extraneous elements from images en route to deriving a mosquito count within a 5% margin of error; that is easily integrated into the GLOBE Observer Mosquito Habitat Mapper tool; and that is ultra-intuitive for the end-user. Some may support ImageJ because the tool has the potential to significantly reduce the time spent on manual counts. However, employing the tool simply for this reason overlooks the importance of the upstream methods required for a successful analysis in ImageJ. Considering this, we recommend, at the most, leveraging ImageJ instead of an exclusive use, as in its current state (version 1.53a) it heavily relies on citizen scientists' sampling technique mirroring that of an expert scientist's. Regarding the next generation of ImageJ (i.e., ImageJ2), it focuses on improving extensibility (a characteristic that facilitates the creation of macros) and interoperability (compatibility with other tools). Although these characteristics are integral to a successful count-automating algorithm, our reservations and recommendations remain since ImageJ2 lacks the optimal quantification features.

Further research is necessary to determine whether the potential benefits of the suggestions outlined herein outweigh the costs of implementation, particularly as it pertains to scientists' prediction of mosquito populations and management of mosquito-borne diseases.

Acknowledgements

NASA, SEES, CSR, TSGC, NESEC – your generous investment in future scientists is inspiring. We are indebted to you. Thank you, program coordinators, for the unique Earth Science experience. The wealth of knowledge and skills gained undoubtedly benefit us as we pursue our endeavors. We would also like to extend our gratitude to the subject matter experts for sharing your knowledge and expertise in areas such as galactic astronomy, geology, and astrobiology. You have greatly enriched our lives. Mosquito Mapper mentors, your support has been invaluable. Special thanks to Dr. Russanne Low, Ms. Cassie Soeffing, Peder Nelson, Pratham Babaria, and Matteo Kimura.

Nathaniel Boateng, lead researcher and collaborative leader for the Mosquito Count Automation team, learned that mosquito ecology impacts human health. He is credited with proposing the research question that culminated into this paper, which explores the feasibility of mosquito count automation in the context of citizen science. He immersed himself in scientific literature to gain an understanding of mosquitoes, ImageJ, and its use in certain science disciplines. His authorship of the Abstract, Introduction, Methods and Materials, Discussion, Conclusion, and Acknowledgements sections reflects the newly acquired knowledge and his propensity to glean insights from experts in the field. Furthermore, his exceptional interest in involving his fellow teammates in the process undoubtedly contributes to the strengths of the arguments herein. The team's slide presentation and poster also bear his imprint. As alluded to earlier, Nathaniel developed and demonstrated leadership skills through his approach to communicating with the team using email, the team's discussion board, meetings, and a shared Google platform, where he posed supplemental research questions for the team's consideration, established project timelines, led meetings, modeled transparency, and consulted mentors to help cement his team's success. Navigating the challenges of group work, his result-driven mindset, collaborative spirit, and role as project lead were welcomed assets for the Mosquito Count Automation team.

Ashwin Roperia contributed to the Methods section and played a crucial role in the use of ImageJ processing for mosquito count automation. This section detailed the experiment conducted, manual count procedure, GLOBE sample data, ImageJ count procedure, and the ImageJ macro. Ashwin learned about how ImageJ is used in the biological science field, and applied that knowledge when he was researching the feasibility of automating the mosquito count using ImageJ. Ashwin is credited with coding the *AutomatedMosquitorCounter* program, an ImageJ macro which makes the ImageJ count procedure more user friendly and streamlined. While coding this script, Ashwin learned about the world of ImageJ plugins and macros which help automate a variety of processes. Ashwin refined and demonstrated his leadership skills as he was given the responsibility of scheduling and hosting Zoom meetings and making sure everyone had the ability to communicate their ideas, which contributed to his team's success. Furthermore, Ashwin also learned how to use iMovie to edit videos and was responsible for editing and uploading the group's video presentation. Ashwin was also an overall reviewer of the team's research paper and poster,

providing high-level commentary and input. His ability to take lead and recommend changes that elevated the level of the project helped cement the team’s success.

Daniela Cabrales, a dependable and strong team-player, enthusiastically contributed to the Conclusion section. The Conclusion section summarizes all the previous sections' thoughts into three clear and concise paragraphs. The conclusion is where all the reader's unanswered questions are resolved. With the knowledge she gained while working on the project, she learned that although there may be an easier and faster way to do things, it is not always the most reliable.

Prayag Sreenivasan contributed to the introduction and hypothesis section. The introduction section acts as a sort of mini-abstract to the rest of the paper, giving readers the general gist of why and how the experiments were conducted. Prayag also contributed to the hypothesis section by proposing and putting down a possible outcome to the central question of the project, as well as some possible implications of it.

Logan Sandell is a masterful narrator who continuously provides exceptional feedback. The power he brought to the Introduction and Experiment sections of our video presentation is unmatched. Equally impressive, he conveyed a strong understanding of our research process during the question and answer period for the SEES Virtual Symposium and in his thorough review of our content.

Micaela Geborkoff, a strong presenter with a commanding voice, elevated the Results section of our video presentation. Her contribution was indeed valuable.

Foluso Osoba performed an overall review of the Google Slides presentation that culminated into our video presentation, as well as provided good high-level commentary of his impressions on ImageJ. His ideas were essential to the team moving forward with its deliverables. He also provided additional images of mosquito traps and environments.

Bibliography

Global Learning and Observations to Benefit the Environment (GLOBE) Program, accessed 21 July 2021, globe.gov

Jaworski, L., Jansen, S., Pfitzner, W. P., Beck, M., Becker, N., Schmidt-Chanasit, J., Kiel, E., & Lühken, R. (2019, July 16). Comparative analysis of subsampling methods for large mosquito samples. Parasites & Vectors. https://parasitesandvectors.biomedcentral.com/ articles/ 10.1186/s13071-019-3606-5.

Little, E., Biehler, D., Leisnham, P. T., Jordan, R., Wilson, S., & LaDeau, S. L. (2017). Socio-Ecological mechanisms supporting high densities of Aedes ALBOPICTUS (DIPTERA: Culicidae) in Baltimore, MD. Journal of Medical Entomology, 54(5), 1183–1192. https://doi.org/10.1093/jme/tjx103

Mains, J. W., Mercer, D. R., & Dobson, S. L. (2008). Digital image analysis to estimate numbers of Aedes Eggs Oviposited in containers. Journal of the American Mosquito Control Association, 24(4), 496–501. https://doi.org/10.2987/5740.1

McClure, E. C., Sievers, M., Brown, C. J., Buelow, C. A., Ditria, E. M., Hayes, M. A., Pearson, R. M., Tulloch, V. J. D., Unsworth, R. K. F., & Connolly, R. M. (2020). Artificial intelligence meets citizen science to supercharge ecological monitoring. Patterns, 1(7), 100109. https://doi.org/10.1016/j.patter.2020.100109

Parker, C., Meador, M., & Hoover, J. P. (2020). Using digital image analysis to QUANTIFY SMALL Arthropod vectors. Journal of Medical Entomology, 57(5), 1671–1674. https://doi.org/10.1093/jme/tjaa072

Rueden, C. T., Schindelin, J., Hiner, M. C., DeZonia, B. E., Walter, A. E., Arena, E. T., & Eliceiri, K. W. (2017, November 29). Imagej2: Imagej for the next generation of scientific image data. BMC Bioinformatics. https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-017-1934-z.