Project Proposal

Vision Guided Mosquito Dissection for the Production of Malaria Vaccine

Nicholas Greene - ngreen29

Mentors: Balazs Vagvolgyi, Alan Lai, Parth Vora

EN.601.656 CIS II Spring 2021

5/6/21

Contents

1 Abstract

The company Sanaria has developed an effective vaccine to combat Malaria, which could be crucial in the fight to eradicate Malaria. A major obstacle preventing large-scale deployment of this vaccine is the cost of it's production. A bottleneck in the production process is the extraction of salivary glands from mosquitos infected with Malaria. Currently, mosquitos must be manually dissected by highly trained workers who extract the salivary glands by hand. The need for skilled labor is very expensive, and it is also very slow. A prototype robotic mosquito dissection system is currently in development at Johns Hopkins which will automate the process of mosquito salivary gland extraction. In this project, four computer vision algorithms are developed to solve various classification and segmentation tasks which are necessary for efficient and robust functioning of the robotic system.

2 Motivation

Malaria is a major global human health issue. In 2019, there were 229 million cases of malaria, with 411 000 deaths [7]. A successful effort to prevent the spread of malaria would have a dramatic impact on the lives of millions of people. Currently, the most successful approach to combating malaria on a large scale is the prevention of mosquito-human contact through various means such as mosquito nets and/or insecticides [6]. These have proved remarkably effective, however the effectiveness and progress has observably plateaued in their in recent years.

Sanaria is a company which has developed the first malaria vaccine which is effective and feasible to produce [2]. Malaria is caused by a parasite which spreads among humans by inhabiting the salivary glands of a host mosquito, which spreads the malaria parasite when it bites humans. The production and distribution of Sanaria's vaccine involves the extraction of these infected mosquito salivary glands. This step during production is a severe bottleneck and is preventing large scale production and deployment of the Sanaria vaccine.

A robotic mosquito dissection system for automating the extraction of infected salivary glands is being developed to make salivary gland extraction significantly more efficient which will enable large scale production of the malaria vaccine [3].

3 Prior Work

The prior work, as outlined in [3], is the mechanical design and software architecture of a robot system which is able to position, decapitate, and extract the salivary glands of mosquitoes in parallel. In addition, a handful of vision algorithms based on both image processing and deep learning have been implemented and are fully integrated with the ROS system. Figure 2 shows a rendering of the robot hardware and the results one of the vision algorithms. The existing hardware and software architecture can accommodate the addition of new camera sensors and vision algorithms without requiring any additional development.

The main design feature of the system is its turntable. The turntable contains many small slots which are perpendicular to the outer edge of the circular turntable. The slots are formed by two walls rising above from the turntable, and are uncovered on all other sides. These these slots get loaded with mosquitoes with the mosquitos during operation. There are multiple processing processing ”stations” located along the outer edge of the turntable. The turntable advances mosquito’s along to each of these stations by rotating. The specific stations are as follows: First is a cutting station where a robotic gripper moves the mosquito into position so that it’s head is removed by a blade. Second is a squeezing station where the mosquito’s salivary glands are squeezed out in a blob of exudate. Finally, a cleaning station, where the mosquito body is washed away from the turntable slot so that a new mosquito can be loaded in.

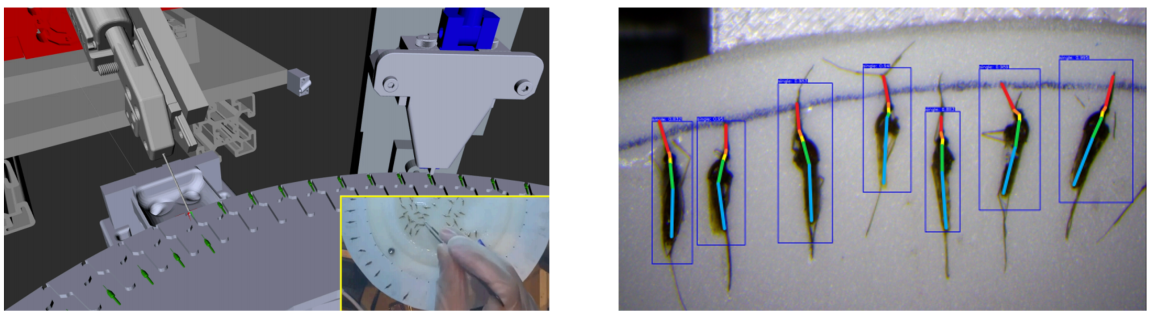

Figure 1: Activities and Deliverables

4 Goals

The goal of this project is to create vision algorithms which will further the development progress of the robotic mosquito dissection system which was described in section 3. These algorithms are important for the robust and efficient performance of the system, for error detection and prevention, and for analysis and evaluation of the system.

The specific vision algorithms this project focuses on are: classification of the turntable cleaning station outcome, classification of the gripper cleaning outcome, and classification and regression of the mosquito exudate volume which is the final product of each mosquito dissection. The turntable cleaning station algorithm determines if there is a mosquito body stuck inside of the turntable slot in order to see if the cleaning station needs to make another attempt at cleaning the slot. The gripper cleaning algorithm determines if any mosquito parts got stuck to the gripper during the cutting stage so that it can be cleaned. Finally, the mosquito exudate volume estimation algorithm first determines if the mosquito was successfully squeezed such that a blob of exudate was extruded correctly. If so, the algorithm then determines the cross-sectional area of the exudate blob which can be used to estimate the volume of the blob.

The The turntable cleaning algorithm has two different implementations, one with traditional image processing, and one with deep learning. The gripper cleaning has a deep learning implementation, and the exudate volume estimation has a classical implementation. In total there are four algorithms.

5 Technical Approach

This section outlines the technical approach for each method and for the system integration.

5.1 Image Processing Methods

5.1.1 Turntable Cleaning Classification

The image processing methods use a combination of color adjustment, filtering, and thresholding, and manually defined regions of interest to accomplish their tasks. Note that the incoming images are in the BGR color format.

The turntable classification algorithm has two stages. One stage segments the mosquito, and the other stage segments specular reflections. The specular reflections must be segmented because they can sometimes end up being counted as part of the mosquito which can interfere with the output. The segmentation method is as follows: Apply a Gaussian blur the input image and extract the region of interest. Convert the result from BGR to HLS color space and then invert the saturation channel. Convert the image back to BGR, and then threshold the image based on BGR Range Values. Finally a median blur is applied. By doing this we are able to segment and ignore some particular reflections which sometimes interfere with the following steps. which particularly look to segment for the mosquito.

The mosquito segmentation stage is as follows: Starting from the blurred region of interest, convert the image from BGR to HSV color space. Set the the Value Channel to it's maximum value for every pixel in the image and then convert the image back to BGR. Threshold the mosquito based on BGR Range Values. Finally, apply a small median blur.

Final steps are to subtract out the segmented specular reflections from the segmented mosquito. Then, if the area of the remaining mosquito pixels is greater than a previsouly defined threshold value, the turntable slot is deemed dirty

This process can be seen in Figure 3

5.1.2 Exudate Classification and Volume Estimation

The algorithm for the exudate detection is broken into classification and segmentation stages. Preprocessing steps befor both stages include the region of interest extration with a predefined binary image mask, and a gaussian blur.

The classification stage's purpose is to determine if there is an exudate blob at all. Using a BGR color based threshold, we can segment the exudate from the background. After the initial color threshold an additional step is performed In order to determine whether or not the currently segmented parts of the image are in fact mosquito exudate, instead of something else like an extraneous mosquito body part. We take advantage of fact that the mosquito parts only show up as thin outlines as a result of the color segmentation from earlier. We apply a median blur with a very large kernel size of 13. Only the densely filled exudate blob's will survive this step. Any features which are thin or sparse are removed. Finally the total area of the remaining pixels after the median blur is compared against a threshold to determine the classification outcome. This can be seen in Figure **??**

If the classification determines that the exudate was squeezed out successfully, then the segmentation stage segments the mosquito pixels. It does so by simlarly using a different

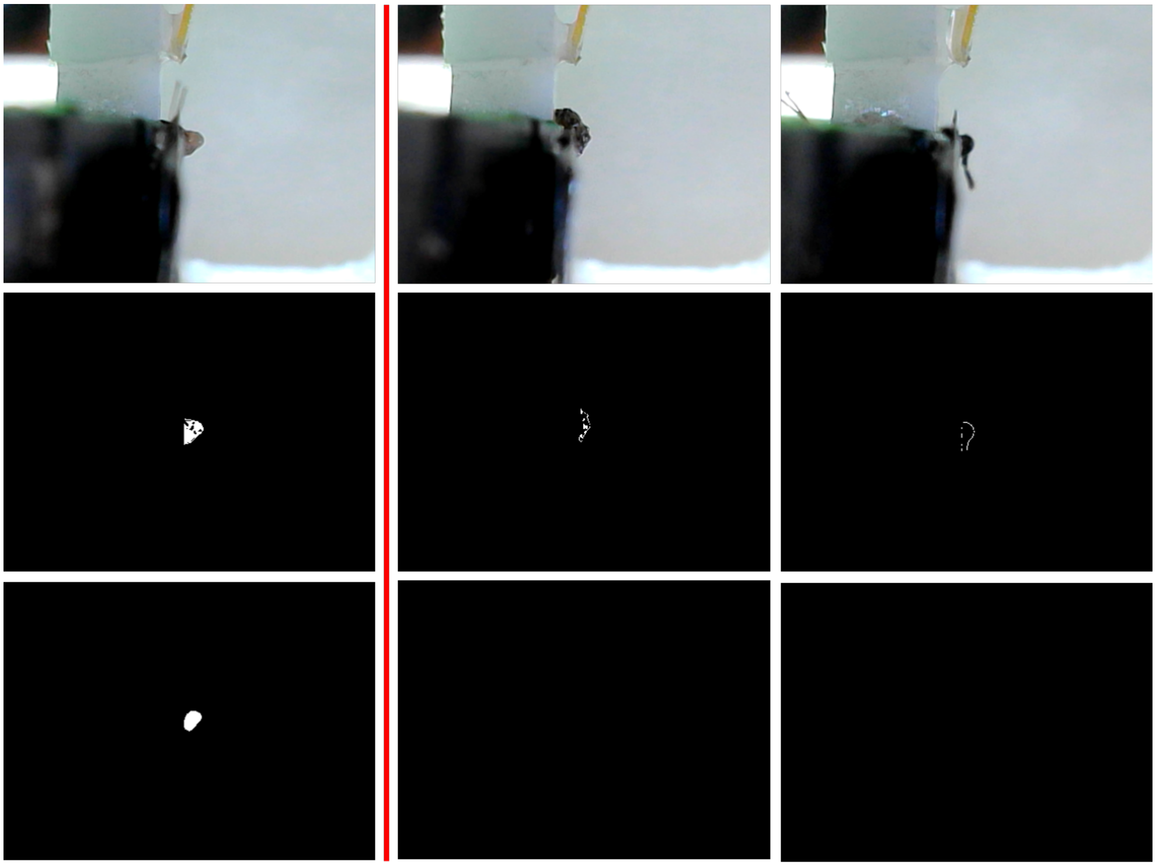

Figure 2: Exudate classification processing steps for the squeezer. The leftmost column was a successful squeeze and the right two columns are failed squeezes. The classifier responds appropriately

color based threshold. This one is a bit less specific than the color threshold from the classification stage. Finally a median blur is used to filter small holes. The area of the segmented blob pixels is returned and this can be used to infer the volume.

5.2 Machine Learning Approach

Considering the small amount of ground truth data available, transfer, about 300 and 600 images for the two datasets, transfer learning seemed to be the most feasible approach to creating machine learning models for the turntable and gripper classification tasks.

5.2.1 Turntable Cleaning Classification

The dataset size for the turntable cleaning task was approximately 300 images. Using transfer learning with VGG16 [5] pretrained on ImageNet [4] where only the fully connected layers at the end are fine-tuned, the performance achieved was 100 on the Test dataset and 98.6% over the entire dataset. The false positive rate was 0.6% and the false negative rate was 0.6% for the whole dataset. Compared to transfer learning with ResNet50 [1] pretrained on ImageNet, VGG16 achieved about 10% better accuracy.

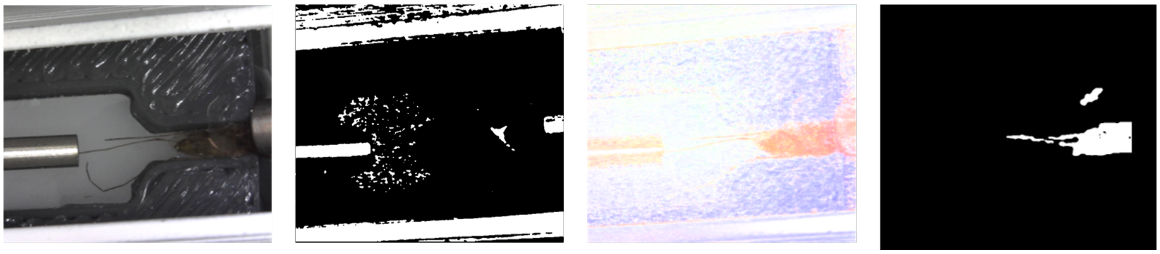

Figure 3: Pipeline for the classical turntable cleaning algorithm. From left to right: input image, segmented reflections, result of setting the HSV value channel to maximum on the input image, final segmentation

5.2.2 Gripper Cleaning Classification

In a very similar manner to the turntable cleaning method, a transfer learning model was used for the gripper cleaning task. The dataset size for this was was just over 600 images. Again VGG16 had the best results with in the same configuration where the fully connected layers at the end were fine tuned. VGG16 outperformed ResNet50 by about 15% for this task. I reached 98.3% accuracy in classification with a 0.6% false positive rate and a 1.05% false negative rate. Figure 4 is an example of the gripper images which the network was trained on.

Figure 4: training images for the gripper cleaning task

5.3 System Integration

In the current layout of the system, there are two computers. The main robot computer interfaces with the actuators and cameras of the robot, while the vision computer, which contains a GPU and which performs the vision tasks, is remote. Integration was completed be done by creating ROS Service wrappers around each image processing and machine learning algorithm. This will allow the robot main computer to call all of the remote vision functions as needed. The flow of data though the system can be seen in figure 5

6 Key Activities and Deliverables

The key activities and deliverables are outlined below in Figure 6. Note that they are categorized into minimum, expected, and maximum tiers. For brevity the chart is condensed;

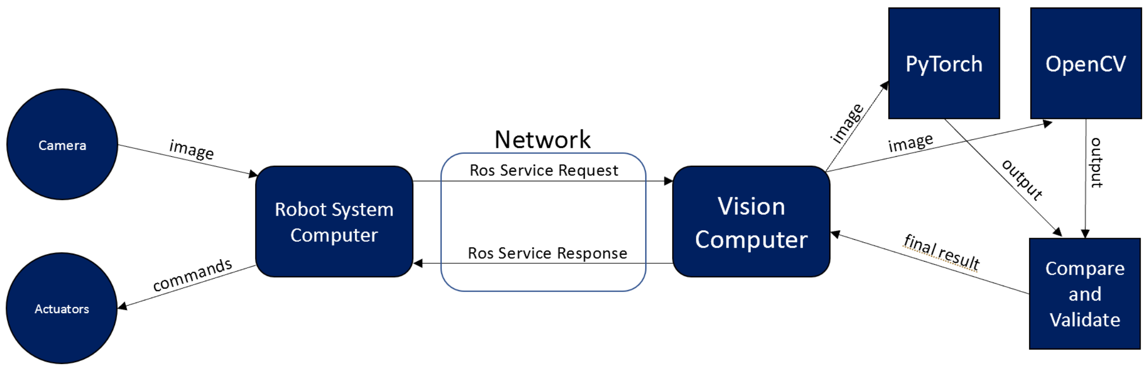

Figure 5: System Overview

the contents of the minimum tier will be repeated for the respective tasks of the expected and maximum tiers.

For each of the three tasks, cleaning station classification, gripper cleaning classification, and exudate estimation, the deliverables consist of the following components:

Machine Learning Approach

- Dataset of 300 hand annotated images
- PyTorch based python code for training a neural network for the task
- PyTorch based python function for quickly evaluating the network's prediction on an image
- ROS service python wrapper for integrating the neural network predition with ROS
- Wiki documentation detailing the network architecture, training procedure, and ROS integration in python

Image Processing Approach

- OpenCV based C++ code of an image processing algorithm for the task
- C++ ROS service wrapper for integrating the algorithm with ROS
- Wiki documentation detailing the image processing algorithm and the ROS integration in C++.

In the end all of the deliverables were reached aside from those associated with an image processing based approach for the gripper cleaning classification task and those associated with a deep learning approach for exudate volume estimation. Specifically I reached all minimum deliverables, half of my expected deliverables and half of my maximum deliverables. The reason I completed half of my maximum deliverables at the expense of fully completing the expected deliverables was because my mentor reccomended I do so. We discussed that the benefit of having a working algorithm for a new task was greater than having an additional working algorithm for the gripper cleaning task. Also, the timing of the hardware development, along with the end of the semester meant it significantly more feasible.

	Activity	Deliverable	Original Expected Deadline	Status
Min	Collect images of cleaning station and annotate	Annotated cleaning station dataset	3/1	Completed
	Implement image processing method for cleaning station	Functioning image processing based code and high quality documentation in a wiki	3/15	Completed
	Develop and train a neural network on the cleaning station dataset	Trained parameters of a neural network classifier, along with code and high quality documentation in a wiki	3/18	Completed
	Integrate both with the rest of the system using ROS	Working ROS services which can be successfully interfaced with from the robot control computer, and thorough documentation for how to use them in a wiki page	3/22	Completed
Expected	Repeat for gripper cleaning	Same deliverables as for turntable cleaning station.	4/7	Completed (Deep Learning) Not Completed (Classical Method)
Max	Discuss with hardware team about collecting exudate data	Plan for collecting exudate ground truth data	4/1	Completed
	Repeat for exudate volume estimation	Same deliverables as for previous two tasks.	5/1	Completed (Classical Method) Not Completed (Deep Learning)

Figure 6: Activities and Deliverables

7 Dependencies

The dependencies, and their respective contingency plans are explained in detail in figure 7. All of my dependencies were met and there were no major issues which completely halted progress. Acquiring images in the very beginning was a bit slower than anticipated but it didn't have a significant impact on the project outcome.

8 Roles and Management

I am the sole group member for this project and my mentors are Balazs Vagvolgyi, Alan Lai, and Parth Vora. Balazs will be my main contact for help with the robot system and with the ROS interfacing. Alan and Parth previously implemented similar machine learning and image processing based vision algorithms which are currently in use. They were a valuable resource for consulting about practical implementations.

The management plan was the following. I attended weekly meetings every with the full mosquito project team where was able to discuss with the other teams, receive feedback, and keep up to date with the status of the project as a whole. I additionally met with my mentor Balazs approximately once every other week on a regular basis to discuss details about the implementations and to and to ask for help.

Dependency	Need	Status	Contingency Plan	Planned Deadline	Hard Deadline
Continued access to GPU	GPU for training neural network	Currently have access to the Diva computer	I have a very capable personal GPU	Ongoing	Ongoing
Cameras for collection of images of each task need to be mounted and integrated	To collect images for annotation	Cameras mounted for turntable cleaning station and squeezing station but not for gripper cleaning	Use existing but less desirable views from the other existing cameras.	3/14	3/22 (for remaining camera)
Hardware team	300+ images of turntable cleaning station in progress with mosquitos.	Hardware team is currently running experiments and collecting these images on a near daily basis while they run their experiments	Begin working on the image processing methods for the other tasks while waiting for additional training data collection.	3/1	3/7
Hardware team	300+ images of gripper cleaning in progress with mosquitos.	The gripper cleaning water jet has not yet been installed, so there are no ongoing gripper cleaning experiments	Small Delay: Begin working "out of order" on the image annotation and image processing for the exudate estimation task Long Delay or Expected Failure: Change from the gripper cleaning task to one of the are many remaining vision tasks	3/14	3/22
Hardware team	Method for ground truth collection of mosquito exudate volume for training images	I will bring this up during the weekly meetings	Abandon neural network approach and only do image processing for this task	3/15	4/1

Figure 7: Dependencies

9 Future Work

I hope to continue working on this project in the future. Regarding future work, there are many more vision tasks which need to be completed for the system. However, all of the methods I implemented have some ways in which they can be improved. The tasks which I solved using classical methods can be improved by creating more robust ways of determining the region of interest. This is largely dependent on the final hardware setup, but will eventually need to be done. Additionally, the Deep learning methods could be trained on more information than just "clean" or "not clean," including the detection of small non-halting but undesirable debris such as mosquito limbs. In fact, provisions are already made in the code to allow for this, and the dataset annotations already include other attributes such as these.

10 Lessons Learned

The most important lesson I learned was that accomplishing a task on a real system with real data takes much much more time at each step than I would have anticipated. I underestimated every task by a significant amount. Another very important lesson that I learned is that using the simplest methods possible at the outset is a good idea. When I initially began working on the first image processing method for the turntable cleaning task, I spent a lot of time stuck in the weeds with overly-complicated pipelines that were not as effective or generalizable as what was achieved with just color threshold and filtering.

11 Reading List

H. Phalen, P. Vagdargi, M. Schrum, S. Chakravarty, A. Canezin, M. Pozin, S. Coemert, I. Iordachita, S. Hoffman, G. Chirikjian, and R. Taylor, “A mosquito pick-and-place system for pfspz-based malaria vaccine production,” IEEE Transactions on Automation Science and Engineering, 2020, issn:1545-5955, doi:10.1109/TASE.2020.2992131.

S. Lathuilière, P. Mesejo, X. Alameda-Pineda and R. Horaud, ”A Comprehensive Analysis of Deep Regression,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 9, pp. 2065-2081, 1 Sept. 2020, doi: 10.1109/TPAMI.2019.2910523.

Lo, Frank Po Wen, Yingnan Sun, Jianing Qiu, and Benny Lo. ”Image-Based Food Classification and Volume Estimation for Dietary Assessment: A Review.” IEEE journal of biomedical and health informatics 24, no. 7 (2020): 1926-1939.

Belagiannis, Vasileios, Christian Rupprecht, Gustavo Carneiro, and Nassir Navab. ”Robust optimization for deep regression.” In Proceedings of the IEEE international conference on computer vision, pp. 2830-2838. 2015.

Chandrarathne, Gayani, Kokul Thanikasalam, and Amalka Pinidiyaarachchi. ”A comprehensive study on deep image classification with small datasets.” In Advances in Electronics Engineering, pp. 93-106. Springer, Singapore, 2020.

Hurtik, Petr, Vojtech Molek, Jan Hula, Marek Vajgl, Pavel Vlasanek, and Tomas Nejezchleba. ”Poly-YOLO: higher speed, more precise detection and instance segmentation for YOLOv3.” arXiv preprint arXiv:2005.13243 (2020).

References

[1] He, K., X. Zhang, S. Ren, and J. Sun (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

[2] Jongo, S. A., S. A. Shekalaghe, L. P. Church, A. J. Ruben, T. Schindler, I. Zenklusen, T. Rutishauser, J. Rothen, A. Tumbo, C. Mkindi, et al. (2018). Safety, immunogenicity, and protective efficacy against controlled human malaria infection of plasmodium falciparum sporozoite vaccine in tanzanian adults. *The American journal of tropical medicine and hygiene 99*(2), 338–349.

[3] Li, W., Z. He, P. Vora, Y. Wang, B. Vagvolgyi, S. Leonard, A. Goodridge, I. Iordachita, S. L. Hoffman, S. Chakravarty, and R. H. Taylor (2021). Automated mosquito salivary gland extractor for pfspz-based malaria vaccine production. *preprint*.

[4] Russakovsky, O., J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV) 115*(3), 211–252.

[5] Simonyan, K. and A. Zisserman (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[6] US Centers for Disease Control and Prevention (2021). "Essential Services for Malaria". [Online]. Available: `https://www.cdc.gov/coronavirus/2019-ncov/global-covid-19/maintain-essential-services-malaria.html`.

[7] World Health Organization (2020). "World Malaria Report 2020". [Online]. Available: `https://www.who.int/publications/i/item/9789240015791`.