Automated Mosquito Salivary Gland Extractor for PfSPZ-based Malaria Vaccine Production

Wanze Li[1], Zhuohong He[1], Parth Vora[1], Yanzhou Wang[1], Balazs Vagvolgyi[1], Simon Leonard[1], Anna Goodridge[1], Iulian Iordachita[1], Stephen L. Hoffman[2], Sumana Chakravarty[2], Russell H. Taylor[1]

***Abstract*— Malaria is a worldwide scourge, and the broad deployment of an effective vaccine would improve the lives of millions of people. A vaccine based on *plasmodium falciparum* (PfSPZ) sporozoites extracted from the salivary glands of infected mosquitoes shows significant promise. However, the large-scale industrial production of PfSPZ-based vaccines will benefit from automation of the key step of extracting sporozoites from mosquito salivary glands that is currently performed by manual microdissection. In this work, we demonstrate a robotic system prototype for extracting salivary glands from mosquitoes to streamline vaccine production and reduce the need for operators. In the proposed system, mosquitoes are decapitated in an automated robotic pick-place-decapitate process, then a squeezer apparatus extracts mosquito salivary glands from the body. Mosquito detection and body part localization are performed by computer vision methods. The software allows system operation in simulation and on the robotic hardware, which facilitates subsystem development and integration. Experiments show encouraging results with success rates of 93% in robotic mosquito manipulation and 87.1% in salivary gland extraction. The system has the potential to improve the efficiency of PfSPZ vaccine production with significant gains in throughput and reduction in training times for a highly deskilled initial manual step. Further, this system is expected to pave the way for a more mature future system.**

***Index Terms*— Malaria Vaccine, Computer Vision, Microrobotics, Soft Object, Automation, Modular Control, ROS, Simulation**

I. Introduction

Malaria is a worldwide scourge. In 2018, the World Health Organization (WHO) reported [1] 228 million cases of malaria infections and 405,000 related deaths worldwide. Furthermore, in 2020, the COVID-19 pandemic significantly slowed health service for other diseases including malaria according to the WHO [2] and the United States Centers for Disease Control and Prevention (CDC) [3]. Current efforts to curb the malaria parasite rely on the use of insecticides to disrupt transmission in mosquitoes and the antibiotic artemisinin to combat the parasite after diagnosis. However, the appearance of resistant mosquitoes and malaria strains in key geographic regions have begun to undermine such efforts. Finding a more permanent solution such as a malaria vaccine has become increasingly urgent.

[1] W. Li, Z. He, P. Vora, Y. Wang, B. Vagvolgyi, S. Leonard, A. Goodridge, I. Iordachita, and R. Taylor are with the Laboratory for Computational Sensing and Robotics at the Johns Hopkins University. [2]S. Hoffman and S. Chakravarty are with Sanaria, Inc. W. Li is the corresponding author, Email: wanzeli@jhu.edu.

This work was supported in part by NIH SBIR grants R43AI112165 and R44AI134500 and in part by Johns Hopkins University internal funds.

Anopheles mosquitoes are insect vectors that infect humans when they introduce the sporozoite (SPZ) stage of the malaria parasite while feeding on blood. Several groups [4]–[9] are developing vaccines for the *plasmodium falciparum* SPZ (PfSPZ) that targets the *plasmodium falciparum* (Pf) parasite which accounts for 99.7% of malaria cases in the WHO African Region in 2018. The PfSPZ vaccine by Sanaria (Sanaria Inc., Rockville, MD, USA) has proven to be a highly effective vaccine with high robustness and long-lasting protective effects. However, large-scale production remains one of the greatest challenges facing the deployment of a PfSPZ-based vaccine. Immense quantities of sporozoites, which reside in the salivary glands of female mosquitoes, are needed to manufacture the vaccine. Gland extraction is a time-consuming and skill-intensive procedure currently performed by a human operator using hand tools and a stereo microscope. To increase dissection efficiency and reduce costs associated with operator training, our team at JHU and Sanaria are developing robotic systems for salivary gland extraction.

We have previously developed a semi-autonomous mosquito micro-dissection system (sAMMS) that can be potentially applied to Sanaria's vaccine manufacturing process [10], [11]. With sAMMS, the salivary gland extraction procedure is simplified into four steps that an operator will execute: mosquito placement, decapitation, squeezing and gland collection. Although sAMMS increased the efficiency of gland extraction from 290 mosquitoes per hour to 450 and greatly reduced operator training time, the procedure still requires significant operator skill and has a "learning curve". A more autonomous approach would address these factors while also improving extraction consistency.

In previous papers [12], [13], we have already documented a prototype that automates certain steps in the dissection procedure as an improvement to the sAMMS system. In this paper, we describe further developments towards a fully-autonomous system that include: (1) a simplified mosquito staging system; (2) a squeezing station for salivary glands extrusion and extraction. While the previous design only automated mosquito pick-and-place, the new system enables continuous dissection of multiple mosquitoes by automating the entire dissection process, including decapitation and salivary gland extraction; (3) improvements to computer vision (CV) algorithms yield higher accuracy and improved robustness for mosquito detection and body part localization; and (4) a new modular system control software based on the Robot Operating System (ROS) [14] that enables automation

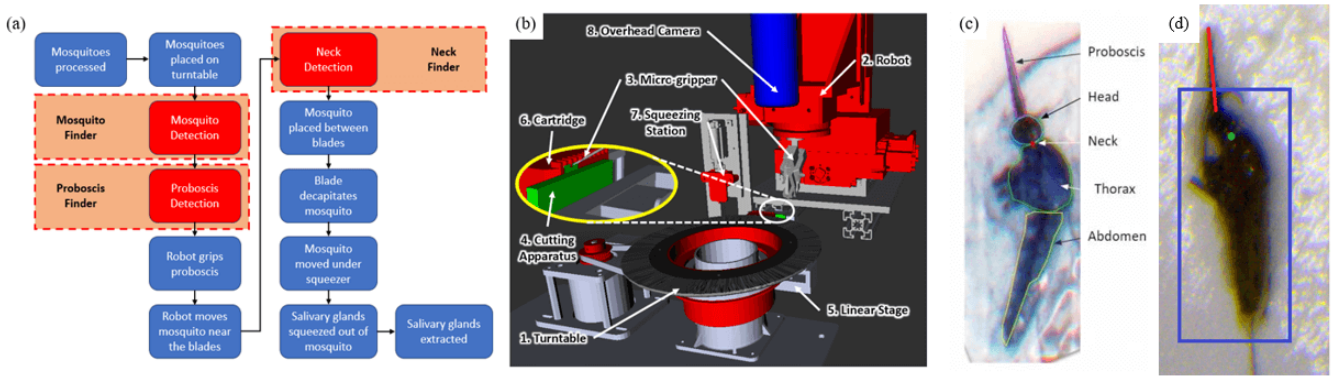

Fig. 1. (a) Flowchart demonstrating key operations in the dissection process. The rounded blue or red boxes represent steps in process. The orange boxes with dashed borders represent CV tasks. (b) A labeled photo of our robotic system in RViz simulation. (c) Parts of a mosquito with labels. (d) Labelled mosquito keypoints. *Mosquito finder* finds the bounding box around the mosquito, in blue. Proboscis and legs are not included in the bounding box. *Proboscis finder* finds the tip and base of the mosquito's proboscis, in red. The angle of the proboscis is defined as the angle of the red line with respect to the vertical axis. *neck finder* identifies the mosquito's neck, in green.

and simulation. Experiments to evaluate the accuracy and robustness of the improved robotic pick-place-decapitate and salivary gland extraction processes show encouraging results and highlight the direction of future development.

II. System Overview

The proposed robot-assisted dissection procedure comprises manual and automated steps. The flow chart shown in Fig. 1a illustrates the major functional steps of the procedure. The first two steps are completed manually by a human operator. The operator first grabs the proboscis (labeled in Fig. 1c) of infected mosquitoes and places them one-by-one on the turntable in a regular radial arrangement. The turntable automatically advances at regular intervals thereby moving the staged mosquitoes toward the robotic subsystem. The robotic subsystem detects mosquitoes that appear on the turntable and carries out the automated processing steps.

The robotic subsystem shown in Fig. 1b consists of eight components: a turntable (1) that transports mosquitoes from the manual loading area to the workspace of a 4 degrees-of-freedom (DoF) Cartesian manipulator robot (2). The role of the robot is to move mosquitoes from the turntable to the cutting apparatus. The four DoFs are translations along the X, Y, and Z axes, and one for controlling the opening of the micro-gripper tool (3) that is mounted as the robot's end-effector. The cutting apparatus (4) decapitates the mosquitoes using a pair of stainless steel blades actuated by a standard SG90 servo motor (Tower Pro Pte Ltd). A linear stage (28HB30L4-05TM) (5) composed of a stepper motor (NEMA 11) and a lead screw drive, transports a 3D printed cartridge (6) from the cutting apparatus to the squeezing station (7). The cartridge moved by the linear stage holds the mosquitoes while the cutting and squeezing steps are performed. The squeezing station features a vertical mounted linear servo (Actuonix PQ12-R) that moves the squeezing pin up and down. Lastly, the overhead camera (OptixCam Summit D3K2-5) (8) monitors the process and provides images to the computer vision (CV) subsystem. The steps of the automated process are illustrated in Fig. 1a. Based on whether the mosquito is manipulated by the robot, the automated process could be divided into two separate parts: mosquito pick-place-decapitate (MPPD) procedure and mosquito salivary gland extraction procedure.

A. Mosquito Pick-place-decapitate (MPPD) Procedure

The MPPD procedure starts with mosquito detection on the turntable. The turntable incrementally rotates the table until the *mosquito finder* CV algorithm detects a mosquito on the image of the overhead camera and calculates the coordinates of its bounding box (see Fig. 2a). When a mosquito is found, the *proboscis finder* CV algorithm detects the location of the proboscis at the top of the bounding box, which the robot controller converts to robot joint coordinates based on an offline robot-to-camera calibration. The robot is then commanded to move the micro-gripper to 6 mm above the position of the proboscis. Next, the gripper is moved down to the height of the turntable's surface and the gripper is closed to grab the proboscis (see Fig. 2b). We use the same gripper mechanism that is described in [12], [13]. The jaws of the gripper are actuated by a cam mechanism driven by a HexTronik HXT900 servo motor.

The robot then drags the mosquito off the turntable and onto the top surface of the adjacent cartridge into the mosquito staging slot near the cutting blades (see Fig. 2c, Fig. 2d). At this point, while holding the mosquito proboscis, the micro-gripper tool-tip is positioned 1.2 mm from the cutting blades and the system needs to determine the exact position of the neck (labeled in Fig. 1c) before aligning the mosquito with the cutting blades in order to achieve accurate decapitation. The CV subsystem takes a new image from the overhead camera and uses the *neck finder* algorithm to locate the mosquito's head (labeled in Fig. 1c) and neck on the image (see Fig. 2e), which the robot controller again converts to robot joint coordinates. The controller then calculates the offset between the cutting blades and the neck's position and commands the robot to move the gripper in a position so that the mosquito's neck is precisely aligned in the cutter (see Fig. 2f and Fig. 2g). The blade then decapitates the

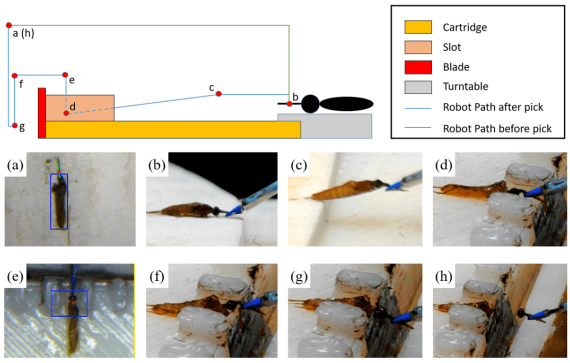

Fig. 2. Side view of the gripper tip path in the MPPD procedure. (a) to (h) are the images when the tool-tip of the micro-gripper at corresponding labeled points. (a) Image of detected mosquito on the turntable before picking with CV algorithms results. (b) Image after grasping the mosquito. (c) Image of the mosquito after being dragged on the surface of cartridge. (d) Image of the mosquito is at the position for neck checking. (e) Image with the results of *neck finder*. (f) Image of the mosquito before placed between blades. (g) Image of the mosquito after placed between blades. (h) Image of the mosquito after decapitation.

mosquito (see Fig. 2h), which enables the removal of the salivary glands from its thorax using the squeezer apparatus. During decapitation the micro-gripper continues to hold the mosquito's proboscis and the attached head. After cutting, the micro-gripper reopens to allow the operator to discard the mosquito head. In our current prototype, the gripper is cleaned manually. The gripper cleaning and head disposal step will be automated in future implementations using a combination of suction and washing.

B. Mosquito Salivary Gland Extraction Procedure

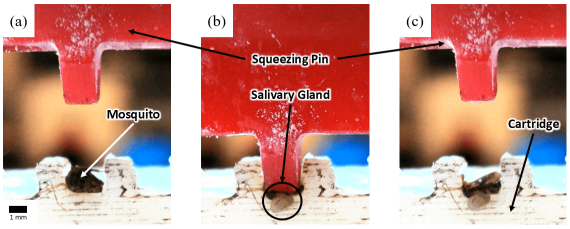

Fig. 3. (a) Front view image of the mosquito on the cartridge before squeezing. (b) Image of the mosquito during squeezing. (c) Image of the mosquito after squeezing.

For salivary gland extraction, the linear stage moves the cartridge so that the cartridge slot containing a decapitated mosquito lines up with the squeezing station. The cartridge is positioned so that the squeezer pin is precisely aligned with the mosquito (see Fig. 3a). Then, the tool moves down and applies pressure on the mosquito's thorax (labeled in Fig. 1c) thereby expelling the salivary glands through the neck (see Fig. 3b). The tool then moves back up and the cartridge moves back to the cutting position (see Fig. 3c). In the current prototype, the collection of extruded salivary glands is not yet implemented. In the future, we plan to add a suction tube that will collect the extruded material. The mosquito remains are currently removed manually by the operator, but this manual task will be replaced by an automated process using a motorized rotating brush in the next version of the system.

In production, the MPPD and salivary gland extraction steps would be performed in parallel. In this case, we expect that the rate limiting step would be dissection, and the cartridge would only advance a small increment so that an empty mosquito staging slot is aligned with the blade in anticipation of the next mosquito to be processed. In the future, a rotary stage would be used for the transport mechanism, and the other apparatus would be arranged to permit parallel operation and increase efficiency. However, our current evaluation focuses on robustness of the individual steps and not on overall processing speed. Accordingly, we currently process mosquitoes one-at-a-time and in small batches in order to facilitate system debugging.

Once the cartridge moves back to the cutting position, the turntable advances until the *mosquito finder* detects a new mosquito, and the dissection process is repeated.

III. System Software Design

A. Robot Controller Design

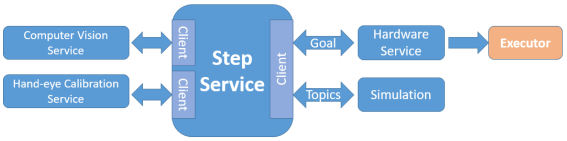

Fig. 4. The block diagram of controller architecture (for one step)

The software controller architecture is designed as an network of services. Each service simply executes a procedure which may entail low level tasks such as commanding a hardware component or high level task such as executing validating a successful mosquito pick. An example of a service is shown in Fig. 4. The communication and synchronization between services in the network rely on a client-server architecture made possible by the ROS actionlib library [14]. At the center of this network is a single high-level client that sends requests to step services according to the sequence of steps shown in Fig. 1a. The proposed architecture also enables a standardized communication of statuses and errors which are useful for handling errors at all levels.

Although ROS actionlib defines a client-service paradigm, it is implemented with messages exchanged through topics instead of ROS services. This implies that any actionlib service request and result exchanged between a client and server uses messages published on ROS topics that can be subscribed or advertised by any ROS node. Our software use this feature to enable the simultaneous use of actionlib-based hardware and simulation nodes as described in Section III-B.

Since the computer vision services return values in pixel coordinates of the overhead camera frame, a hand-eye calibration is required to find the relationship between the coordinates in the camera frame and the joints of the robot. The calibration procedure involves fitting a polynomial

relating the image coordinates to the coordinates of the robot. This calibration method is preferred over methods based on special Euclidean group because of the difficulty of manipulating calibration checkerboards with the hardware. In the calibration procedure, the robot moves the gripper through 100 grid points over a 15 mm $\times$ 30 mm rectangular area that coincide with the surface of the turntable. The robot stops at each grid point, where the tip of the gripper is detected in the image and its pixel coordinates are recorded. Then, a Bernstein polynomial fitting, as displayed in [15], is applied to map the tip coordinates in the camera frame to the robot encoder positions. After the calibration, the image coordinates returned by the computer vision algorithm can be readily converted to robot joint positions.

B. Simulation

In order to aid controller and software development amidst the COVID-19 lockdown, we built a simulation platform that simulates all the components, processes, and behaviors of our physical system. The goal is to allow researchers to facilitate the integration and evaluation of the components described in Section II without access to the physical robot in the lab. The simulation platform relies on ROS for inter-process communication, the Robot Visualization tool (RViz) for 3D visualization and RQt for Graphical User Interface (GUI). Each hardware component in the simulation - turntable, robot, micro-gripper, cutting apparatus, linear stage, and squeezing station - is operated by a separate component simulator. The appearance and kinematic configuration of each component is defined in a Unified Robot Description Format (URDF) file.

Fig. 5. (a) Working flowchart for a component simulator. Separate simulators exists for the turntable, robot, gripper, cutting aparatus, linear stage, and squeezing station. (b) Screenshot of the GUI used to manually control the simulation.

As simulator flowchart (Fig. 5) shows, the software for each component simulator consists of two required ROS nodes and a third optional node. The first required node is the simulator interface, which handles communication with the robot control software and breaks down high-level robot action commands into one or more consecutive joint move commands. These commands are then sent to the other required node: the trajectory generator, which generates a time sequence of joint positions in response to each linear joint move command. Joint configuration, velocity and acceleration limits for trajectory generation are defined in the URDF files. The third, optional node is needed for hardware components that have non-linear kinematics or multiple moving parts (sub-components) actuated by a single degree-of-freedom drive, such as cams or belt-driven gears. In such cases, the joint converter node is used to simulate the component's kinematics by converting the input linear joint position to one or more linear or non-linear joint positions for the sub-components. The final joint states are visualized in RViz. After the motion is completed, a status message is sent back to the interface. Meanwhile, the RQt-based GUI in Fig. 5b allows for manual control of the simulation.

The simulator is capable of simulating hardware and communication errors based on error probabilities specified by the controller. Currently three types of errors are supported: warnings, halting and fatal errors. Warnings signal that the component experienced something irregular but is still able to complete the motion with possible delays. Halting errors indicate that the motion command can not be completed for some reason. Fatal errors are thrown when a component is unreachable due to, for example, a power outage or disconnected wires. Errors generated by trajectory generators appear on the interface's *result* ROS topic for the robot controller to handle. Simulating errors allow us to develop robust control algorithms that are able to mitigate an array of hardware and software errors, especially critical and deleterious errors that may have a low probability of occurring naturally.

High-fidelity simulation featuring accurate visualization and error generation enables our team to continue system development and integration even with limited access to physical robot hardware.

IV. Computer Vision Methods

A. Overview

We have developed several computer vision algorithms to provide robust guidance to the automated dissection system. Our algorithms identify mosquito key points, including the head, proboscis, and neck, to assist in the MPPD process, as illustrated in Fig. 1a and Fig. 1d. Specifically, algorithms detect mosquitoes on the turntable, identify their proboscis and locate the neck that enables accurate alignment of mosquitoes in the cutter. Individual algorithms were evaluated by comparing algorithm output to hand-labelled ground truth for over 250 mosquitoes on turntables. During evaluation, automated scripts processed images in the evaluation dataset and stored output labels and positions in files, which were then processed to determine the number of mosquitoes that the algorithms found within an acceptable margin of error.

B. Mosquito Finder

The mosquito finder algorithm returns a bounding box around mosquitoes on the turntable within a given region of interest (ROI). First, the ROI is extracted, converted to grayscale, and binary inverse thresholded to identify

mosquito pixels. Next, successive iterations of erosion and dilation are performed and connected components are identified. Connected components are filtered by size, and bounding boxes around these components are calculated and returned. Mosquito finder performance was measured using the Intersection over Union (IoU). A mosquito was considered found if the mosquito finder algorithm successfully returned a bounding box around the mosquito with an IoU greater than 0.4. Accuracy of mosquito finder was 95%, with a mean IoU score of 0.736.

C. Head Finder

The head finder algorithm identifies the center of the head of a mosquito. The algorithm takes an image and ROI, and performs template matching with several templates of mosquito heads within the ROI. The algorithm accepts the best match above a pre-set threshold, calculates the midpoint of the mosquito head, and returns the location of the mosquito head midpoint. The head finder was not evaluated separately, since it is a subroutine for proboscis finder and neck finder, which were evaluated.

D. Proboscis Finder

The proboscis finder algorithm identifies the base and tip of a mosquito's proboscis. The algorithm first uses the head finder algorithm to find the midpoint of the mosquito's head. The angle of the proboscis with respect to the vertical is then calculated by warping a region around the head to polar coordinates and searching for the row with the lowest intensity values. The base of the proboscis defined as the point on the circumference of the mosquito head that lies on the angle of the proboscis. The tip is defined as a point a fixed distance away from the base along the same angle as the proboscis. These points are converted to Cartesian coordinates and returned. Accuracy of proboscis finder was 96%, with a mean root mean squared lateral error of 3.46 pixels.

E. Neck Finder

The neck finder algorithm identifies the neck of the mosquito. The algorithm first uses the head finder algorithm to find the midpoint of the mosquito's head. The neck is defined as the point on the circumference of the mosquito's head directly below the center, which is calculated by the algorithm and returned. Accuracy of neck finder was 95%, with a mean root mean squared vertical error of 10.91 pixels.

V. Evaluation

A. Experiment Setup

Testing was performed with 100 non-infected mosquitoes to investigate the efficacy and stability of the system. Prior to the experiment, mosquitoes were stored in an airtight container of phosphate-buffered saline solution (PBS) at 35-40 °F in order to keep the insects fresh and soft. The experiment setup is shown in the Fig. 6; the entire system was setup on an optical table. The robot was controlled by a Galil controller (DMC-4143) that communicated with a Linux computer via ethernet. The linear stage, cutting apparatus and squeezing station were controlled by one Arduino Uno microprocessor. The turntable and micro-gripper were controlled by two other Arduino Unos. All three Uno components received commands from the computer via USB serial communication.

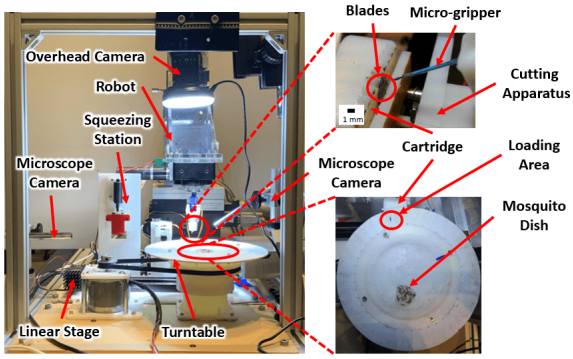

Fig. 6. Front view image of the experiment setup with close-up image of cutting area and turntable.

Four cameras were used for data collection. The system overhead camera (OptixCam Summit D3K2-5) and an Opti-Tekscope USB Microscope camera mounted near the cutting apparatus were used to visualize the MPPD procedure. Two other Opti-Tekscope USB Microscope cameras were mounted on the front and lateral sides of the squeezing station to visualize the salivary gland extrusion process for the observer who recorded whether a trial succeeded or not. All actions of the MPPD and the salivary extrusion procedure were fully autonomous. An MPPD outcome was considered successful only if the decapitated head was detected. An extrusion result was considered successful only if extruded material was detected outside the body.

Before starting the automated dissection process, the operator placed dead mosquitoes in a shallow dish filled with PBS located in the center of the turntable, as shown in Fig. 6, then grabbed mosquitoes one-by-one by their proboscis using handheld micro-tweezers and placed them on the loading area of the turntable in a regular radial arrangement. Once the mosquitoes were set up on the turntable, the operator started the automated dissection process while observing and recording the results.

B. Evaluation of Mosquito Pick-Place-Decapitate (MPPD) Procedure

The MPPD procedure achieved a 93% success rate, outperforming our previously reported rate of 90% [12]. Quantitative experimental results are shown in Table I. Of the 100 mosquitoes tested, 93 were successfully picked from the turntable and placed in a proper position between the blades. Of the properly placed mosquitoes, the blades then decapitated all 93 mosquitoes which result in an the MPPD success rate of 93%, demonstrating the effectiveness of the computer vision, control and hardware.

6 out of 7 failures in the Pick-Place step were the result of inaccurate computer vision detection. Fig. 7a shows a

TABLE I
QUANTITATIVE EVALUATION RESULTS

Procedure	Step	Success	Failure	Total	Success Rate
MPPD	Pick-Place	93	7	100	93%
	Decapitate	93	0	93	100%
Gland Extract	Squeezing	81	12	93	87.1%
Overall		81	19	100	81%

reference image of an accurate neck position detection. Three of six CV failures were caused by the inaccurate positions provided by the *neck finder* algorithm (see Fig. 7b), which yielded inaccurate neck-to-tooltip offsets leading to failed mosquito placement in the cutting blades as shown in Fig. 7c. *neck finder* also failed to detect the neck at all for two times which caused two of the three remaining CV failures. The other CV failure was caused by the *proboscis finder*, which is illustrated in Fig. 7d. Proboscis finder and neck finder failed primarily when template matching to find the head of the mosquito failed. This could be rectified by using a selection of templates that is better optimized to cover the possible mosquito head shape appearances.

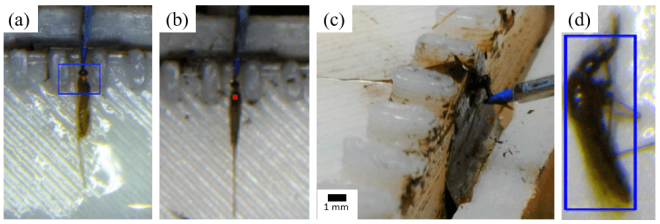

Fig. 7. Computer vision failure modes: (a) Image of mosquito with successful detection by *neck finder*; neck position marked with red dot. (b) Example of failed neck detection by *neck finder*; red dot marks inaccurate position on thorax. (c) Inaccurate neck detection leads to misalignment in cutting blade; blade is aligned with thorax. (d) Example of failed proboscis detection; *proboscis finder* could not identify proboscis on image.

C. Evaluation of Salivary Gland Extraction Procedure

Salivary gland extraction occurs sequentially after MPPD. Only the 93 mosquitoes that were decapitated correctly were used in this experiment. Among these mosquitoes, 81 had their gland successfully extruded by the squeezing station resulting in a 87.1% success rate.

The majority of the failed extractions (10 out of 12) were attributed to a similar reason. During the some decapitations, the blade motion pulled mosquito bodies in actuation direction and pushed the thorax above the blades (Fig. 8b). As a result, the mosquito thorax was found in an unfavorable position between the blades after decapitation (Fig. 8c). As the cartridge moved towards the squeezing station, the relative motion between the blades and cartridge brought the unfavorably placed mosquito thorax farther forwards, over the edge of the cartridge (Fig. 8d). During the following gland extrusion step, the squeezing pin contacted the abdomen, instead of the thorax, which resulted in a failed gland extraction (Fig. 8e and f). We found several possibilities for to explain the problem scenario including variations in mosquito stiffness due to the onset of rigor mortis, the orientation of mosquitoes in the slot, CV *neck finder* algorithm inaccuracy, and blade dullness. We will conduct isolated experiments to further understand and mitigate the problem.

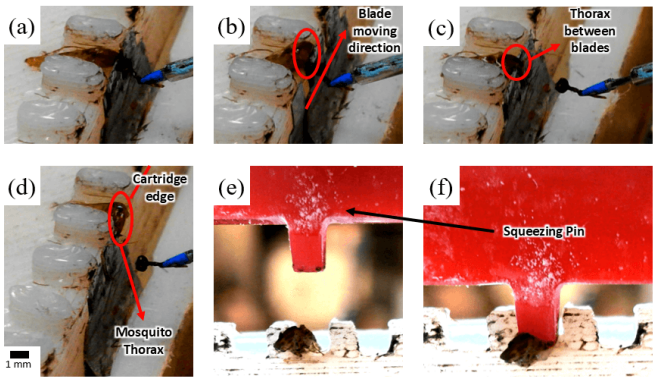

Fig. 8. Recurring mosquito decapitation issue: (a) Mosquito's neck correctly aligned with blade in cutting position. (b) On blade actuation decapitation succeeds but thorax gets pushed upwards. (c) Mosquito's thorax is resting on top of blades in an unfavorable position. (d) As cartridge moves to squeezer, thorax is caught in blade and pulled forward. (e) Incorrectly aligned mosquito before squeezing and (f) after squeezing.

D. Mosquito Preparation

During system development, we realized the impact of proper mosquito preparation on the effectiveness of robotic dissection. Lengthy storage in PBS significantly increased the body stiffness of mosquitoes, adding an uncertainty to the procedure. In our experience, the stiffness interfered with the successful placement of mosquitoes for decapitation and caused a reduction in extracted volume of salivary glands.

VI. CONCLUSION AND FUTURE WORK

In this paper, we report the development of a prototype autonomous robot system that implements several procedures including mosquito pick-and-place, decapitation, and salivary gland extrusion, as part of a larger effort to automate a key step in the industrial production of PfSPZ vaccines. Experimental results indicating the accuracy and robustness of our current system are encouraging, although more work remains to be done. In the near future, we are considering alternative designs for the mosquito transport mechanism and other components to improve throughput and applicability. Furthermore, we are investigating new accuracy improvements to our computer vision methods and are implementing computer vision methods to verify the correct execution of each step in the process. Also, additional methods for autonomous error detection and recovery will be integrated to the software controller, and we will begin investigation of methods to optimize system throughput using parallelism and careful tuning of each component. Lastly, our experiments have highlighted the need to develop an improved method for preparing mosquitoes for dissection that results in more consistent mosquito body stiffness while preserving the freshness of salivary glands.

REFERENCES

[1] World Health Organization, "World malaria report 2019," Dec 2019. [Online]. Available: https://www.who.int/publications-detail/world-malaria-report-2019

[2] World Health Organization, "COVID-19 significantly impacts health services for noncommunicable diseases," Jan 2020. [Online]. Available: https://www.who.int/news-room/detail/01-06-2020-covid-19-significantly-impacts-health-services-for-noncommunicable-diseases

[3] United States Centers for Disease Control and Prevention, "Maintaining Essential Health Services During COVID-19 in Low Resource, Non-U.S. Settings," Nov 2019. [Online]. Available: https://www.cdc.gov/coronavirus/2019-ncov/global-covid-19/essential-health-services.html

[4] B. Mordmüller, G. Surat, H. Lagler, S. Chakravarty, A. S. Ishizuka, A. Lalremruata, M. Gmeiner, J. J. Campo, M. Esen, A. J. Ruben, J. Held, C. L. Calle, J. B. Mengue, T. Gebru, J. Ibáñez, M. Sulyok, E. R. James, P. F. Billingsley, K. C. Natasha, A. Manoj, T. Murshedkar, A. Gunasekera, A. G. Eappen, T. Li, R. E. Stafford, M. Li, P. L. Felgner, R. A. Seder, T. L. Richie, B. K. L. Sim, S. L. Hoffman, and P. G. Kremsner, "Sterile protection against human malaria by chemoattenuated PfSPZ vaccine," *Nature*, vol. 542, no. 7642, pp. 445–449, Feb 2017. [Online]. Available: https://doi.org/10.1038/nature21060

[5] T. L. Richie, P. F. Billingsley, B. K. L. Sim, E. R. James, S. Chakravarty, J. E. Epstein, K. E. Lyke, B. Mordmüller, P. Alonso, P. E. Duffy, O. K. Doumbo, R. W. Sauerwein, M. Tanner, S. Abdulla, P. G. Kremsner, R. A. Seder, and S. L. Hoffman, "Progress with Plasmodium falciparum sporozoite (PfSPZ)-based malaria vaccines," *Vaccine*, vol. 33, no. 52, pp. 7452–7461, Dec 2015, 26469720[pmid]. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26469720

[6] I. Zenklusen, S. Jongo, S. Abdulla, K. Ramadhani, B. Sim, H. Cardamone, E. Flannery, T. Nguyen, M. Fishbaugher, R. Steel, W. Betz, N. Carmago, S. Mikolajczak, S. Kappe, S. Hoffman, B. Sack, and C. Daubenberger, "Immunization of malaria-preexposed volunteers with pfspz vaccine elicits long-lived igm invasion-inhibitory and complement-fixing antibodies," *The Journal of infectious diseases*, vol. 217, May 2018.

[7] R. A. Seder, L.-J. Chang, M. E. Enama, K. L. Zephir, U. N. Sarwar, I. J. Gordon, L. A. Holman, E. R. James, P. F. Billingsley, A. Gunasekera, A. Richman, S. Chakravarty, A. Manoj, S. Velmurugan, M. Li, A. J. Ruben, T. Li, A. G. Eappen, R. E. Stafford, S. H. Plummer, C. S. Hendel, L. Novik, P. J. M. Costner, F. H. Mendoza, J. G. Saunders, M. C. Nason, J. H. Richardson, J. Murphy, S. A. Davidson, T. L. Richie, M. Sedegah, A. Sutamihardja, G. A. Fahle, K. E. Lyke, M. B. Laurens, M. Roederer, K. Tewari, J. E. Epstein, B. K. L. Sim, J. E. Ledgerwood, B. S. Graham, S. L. Hoffman, and , "Protection against malaria by intravenous immunization with a nonreplicating sporozoite vaccine," *Science*, vol. 341, no. 6152, pp. 1359–1365, 2013. [Online]. Available: https://science.sciencemag.org/content/341/6152/1359

[8] M. S. Sissoko, S. A. Healy, A. Katile, F. Omaswa, I. Zaidi, E. E. Gabriel, B. Kamate, Y. Samake, M. A. Guindo, A. Dolo, A. Niangaly, K. Niaré, A. Zeguime, K. Sissoko, H. Diallo, I. Thera, K. Ding, M. P. Fay, E. M. O'Connell, T. B. Nutman, S. Wong-Madden, T. Murshedkar, A. J. Ruben, M. Li, Y. Abebe, A. Manoj, A. Gunasekera, S. Chakravarty, B. K. L. Sim, P. F. Billingsley, E. R. James, M. Walther, T. L. Richie, S. L. Hoffman, O. Doumbo, and P. E. Duffy, "Safety and efficacy of pfspz vaccine against plasmodium falciparum via direct venous inoculation in healthy malaria-exposed adults in mali: a randomised, double-blind phase 1 trial," *The Lancet Infectious Diseases*, vol. 17, no. 5, pp. 498 – 509, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1473309917301044

[9] J. E. Epstein, K. Tewari, K. E. Lyke, B. K. L. Sim, P. F. Billingsley, M. B. Laurens, A. Gunasekera, S. Chakravarty, E. R. James, M. Sedegah, A. Richman, S. Velmurugan, S. Reyes, M. Li, K. Tucker, A. Ahumada, A. J. Ruben, T. Li, R. Stafford, A. G. Eappen, C. Tamminga, J. W. Bennett, C. F. Ockenhouse, J. R. Murphy, J. Komisar, N. Thomas, M. Loyevsky, A. Birkett, C. V. Plowe, C. Loucq, R. Edelman, T. L. Richie, R. A. Seder, and S. L. Hoffman, "Live attenuated malaria vaccine designed to protect through hepatic cd8+ t cell immunity," *Science*, vol. 334, no. 6055, pp. 475–480, 2011. [Online]. Available: https://science.sciencemag.org/content/334/6055/475

[10] M. Schrum, A. Canezin, S. Chakravarty, M. Laskowski, S. Comert, Y. Sevimli, G. S. Chirikjian, S. L. Hoffman, and R. H. Taylor, "An efficient production process for extracting salivary glands from mosquitoes," 2019, arXiv:1903.02532 [q-bio.QM]. [Online]. Available: https://arxiv.org/abs/1903.02532

[11] R. H. Taylor, A. Canezin, M. Schram, I. Iordachita, G. Chirikjian, M. Laskowski, S. Chakravarty, and S. Hoffman, "Mosquito Salivary Gland Extraction Device and Methods of Use," Patent 20 170 355 951, December, 2017. [Online]. Available: https://www.freepatentsonline.com/y2017/0355951.html

[12] 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.

[13] H. Phalen, P. Vagdargi, M. Pozin, S. Chakravarty, G. S. Chirikjian, I. Iordachita, and R. H. Taylor, "Mosquito pick-and-place: Automating a key step in pfspz-based malaria vaccine production," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 12–17.

[14] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. B. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, 2009.

[15] L. Feng, P. Wilkening, Y. Sevimli, M. Balicki, K. Olds, and R. Taylor, "Accuracy assessment and kinematic calibration of the robotic endoscopic microsurgical system," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2016*, vol. 2016-October. Institute of Electrical and Electronics Engineers Inc., Oct. 2016, pp. 5091–5094, 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2016 ; Conference date: 16-08-2016 Through 20-08-2016.