

Supporting Smartphone-Based Image Capture of Rapid Diagnostic Tests in Low-Resource Settings

Chunjong Park*, Alex Mariakakis*, Jane Yang[†], Diego Lassala[†], Yasamba Djiguiba[†], Youssouf Keita[†], Hawa Diarra[†], Beatrice Wasunna[◦], Fatou Fall[◦], Marème Soda Gaye[◦], Bara Ndiaye[◦], Ari Johnson[‡], Isaac Holeman^{•*}, Shwetak Patel^{*}

University of Washington*, Seattle, WA, USA
{cjparkuw, atm15, shwetak}@cs.washington.edu, iholeman@uw.edu

Muso, [†]Bamako, Mali and [‡]San Francisco, CA, USA
{jyang, dlassala, ydjiguiba, ykeita, ajohnson}@musohealth.org, drhawadiarra@gmail.com
Medic Mobile, [◦]Nairobi, Kenya and [•]San Francisco, CA, USA
{beatrice, fatou, mareme, bara, isaac}@medicmobile.org

ABSTRACT

Rapid diagnostic tests (RDTs) provide point-of-care medical diagnosis without sophisticated laboratory equipment, making them especially useful for community health workers (CHWs). Because the procedure for completing a malaria RDT is error-prone, CHWs are often asked to carry completed RDTs back to their supervisors. Doing so makes RDTs susceptible to deterioration and introduces inefficiencies in the CHWs' workflow. In this work, we propose a smartphone-based RDT capture app, RDTScan, that facilitates the collection of high-quality RDT images to support CHWs in the field. RDTScan does not require an external adapter to control the image capture environment, but instead provides real-time guidance using image processing to obtain the best image possible. During our evaluation study, we found that RDTScan had 98.1% sensitivity and 99.7% specificity against visual inspection of the RDTs. RDTScan helped CHWs capture high-quality RDT images within 18 seconds while enabling a better RDT workflow.

CCS CONCEPTS

- Human-centered computing → Smartphones;
- Applied computing → Consumer health;
- Computing methodologies → Image processing.

KEYWORDS

Rapid diagnostic tests (RDTs); malaria; mobile health; image processing; image quality control

ACM Reference Format:

Chunjong Park, Alex Mariakakis , Jane Yang , Diego Lassala , Yasamba Djiguiba , Youssouf Keita , Hawa Diarra , Beatrice Wasunna , Fatou Fall , Marème Soda Gaye , Bara Ndiaye , Ari Johnson , Isaac Holeman , and Shwetak Patel . 2020. Supporting Smartphone-Based Image Capture of Rapid Diagnostic

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ICTD '20, June 17–20, 2020, Guayaquil, Ecuador

© 2020 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8762-0/20/06.

<https://doi.org/10.1145/3392561.3394630>



Figure 1: RDTScan helps community health workers capture high-quality images of malaria rapid diagnostic tests (RDTs) collected in real-world environments without the need for extra hardware.

Tests in Low-Resource Settings. In *Information and Communication Technologies and Development (ICTD '20), June 17–20, 2020, Guayaquil, Ecuador*. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3392561.3394630>

1 INTRODUCTION

Rapid diagnostic tests (RDTs) offer a convenient way to diagnose various infectious diseases in point-of-care settings [22]. By simply drawing a small amount of blood or swabbing the nose or mouth, a healthcare provider can apply a biological sample to an immunoassay test and see a test result in the form of a colored test line. In this work, we focus our attention on malaria, which is responsible for 24% of all deaths among Malian children under the age of 5 [26]. RDTs provide a low-cost (~\$1 USD) alternative to microscopy for the diagnosis of malaria, playing a crucial role in expedited treatment and global health efforts. In Western Uganda, for instance, one study found that RDTs more than tripled the detection rate of malaria while reducing patients' average clinical visit duration and their odds of being referred to another clinic [3].

The procedure for completing an RDT typically requires many steps: checking the test's expiration date, drawing blood, putting the correct amount of blood and buffer solution into the correct wells, waiting for the prescribed time for the strip to activate, examining the test area for a control line that indicates the test was done properly, and then finally reading the test line itself. Each of these steps is simple to complete, allowing community health workers (CHWs) who may not have the same medical expertise as a physician to still provide vital health services in the field; however, these steps are not always trivial to complete *properly*. Facing time pressure and triaging clinical priorities, CHWs may add the wrong amount of blood to the RDT or not wait the correct duration to read the lines [25]. Inexperienced CHWs may also misinterpret the lines in the result window, either missing faint lines or imagining lines that do not exist [12, 14, 23]. For these reasons, CHW supervisors will often ask CHWs to bring the completed RDTs back to the clinic to ensure the correct decision was reached. Due to operational constraints limiting CHWs' ability to travel frequently to the clinic, days or weeks can elapse until an RDT is actually reviewed.

Smartphones are already making a significant impact in the way that healthcare is distributed in low- and middle-income settings [15], and researchers have sought to use smartphone cameras to interpret RDT results on the behalf of healthcare providers [5, 6, 17, 19]. However, these efforts have typically required additional hardware to control the position of the RDT relative to the camera and the lighting environment, limiting the scalability of their solutions. A mobile, adaptable solution to RDT capture becomes increasingly relevant, as more healthcare moves outside of controlled laboratory and clinic settings and into the community.

We propose RDTScan, a smartphone app that aids CHWs in their workflow without the need for additional hardware. Instead of requiring CHWs to transport RDTs back to their supervisors for review, RDTScan enables them to capture high-quality RDT images that lead to comparable interpretation results relative to examining the physical RDTs themselves. RDTScan is designed to operate on devices with limited processing power and battery life to accommodate the various smartphones that may be deployed in these settings.

To evaluate RDTScan, we first compared the test results that trained lab technicians read from images against those that were read from the RDTs themselves. We found that the readings from RDTScan's images resulted in 99.7% specificity and 98.1% sensitivity. We then deployed RDTScan for 3 months with a non-governmental organization in Mali to evaluate the app's usability and utility. We showed that CHWs were generally successful and improved over time when using RDTScan. After 20 trials, CHWs were able to automatically capture an image 78% of the time with an average capture time of 18 seconds, and only 7.6% of those images were not suitable for review by a supervisor. Through semi-structured interviews, CHWs and their supervisors noted that RDTScan was useful for documenting RDTs in high-quality images for review.

Our research contributes:

- (1) RDTScan, a smartphone app that guides CHWs through capturing high-quality RDT images,
- (2) Quantitative findings from a semi-controlled evaluation with lab technicians on the quality of the captured images,

- (3) Quantitative and qualitative findings from a three-month deployment of RDTScan in Mali.

2 RELATED WORK

Before describing RDTScan, we provide an overview of the literature regarding how RDTs are currently used in the field. We then examine past proposals for improving RDT practices through the use of technology and define where RDTScan falls within this space.

2.1 Current Malaria Diagnostic Practices

Giems stain microscopy and RDTs are the two predominant malaria diagnostic methods that are used in the field. Giems stain microscopy [28] entails examining a thin film of blood under a microscope to look for parasites. Microscopy is a desirable method for malaria diagnosis because it allows healthcare providers to quantify and differentiate malaria parasites, but it requires a well-trained microscopist [27]. RDTs do not require such expertise. As long as the healthcare provider is able to draw blood, place liquids in different cassette locations, and read colored lines against a white background, they are hypothetically able to use an RDT.

Azikiwe et al. [1] compared RDTs against Giems stain microscopy and found that RDTs had comparable specificity and higher sensitivity than microscopy; however, it should be noted their tests were conducted by experts in malaria diagnostics. Harvey et al. [10] investigated the ability of non-expert CHWs to both comply with the RDT procedure and interpret the results of the RDTs. Harvey et al. found that only 57% of their participants were able to perform the procedure correctly and 54% were able to interpret the test results correctly, with the most common mistakes being failures to read faint positive lines or identify invalid results. Those numbers were boosted to 92% and 93%, respectively, when the CHWs were enrolled in a multi-day training program. Such training programs are not widespread in practice since they can take CHWs away from other responsibilities. Thus, our work seeks to improve current malaria diagnostic practices by providing CHWs with real-time guidance while photographing RDTs.

2.2 Automated RDT Capture and Interpretation

Image processing has been proposed as a way of either ensuring that a high-quality RDT photograph is captured or replacing subjective visual interpretation with objective calculations. Image processing approaches often struggle in real-world scenarios when the imaging environment is unconstrained [2]. Unknown positioning between the camera and the object, uncontrolled ambient lighting, and occlusion are just some of the issues that arise when people are asked to take pictures of objects outside of laboratory settings. Researchers have overcome these issues in the past using physical adapters (Figure 2). The Deki Reader¹ is a battery-operated device that provides an enclosed chamber where a healthcare provider can place an RDT to block out ambient lighting; meanwhile, a smartphone-like reader rests over the chamber to provide lighting and fix the position between the RDT and the camera. Herrera et al. [11] found that the

¹<http://fio.com/>



Figure 2: Two systems that facilitate RDT capture and interpretation through additional hardware: (left) Mudanyali et al. [17] and (right) Dell et al. [5, 6].

Deki Reader had 99% concordance with expert visual interpretation. Mudanyali et al. [17] developed a more compact smartphone attachment similar to the Deki Reader. Their attachment includes 3 LED arrays that provide uniform illumination on the RDT, and a convex lens enhances the resolution of the images. Ozkan et al. [19] and Dell et al. [5, 6] both explored attachments that fix the position of the RDT relative to a camera but allow for ambient illumination.

There have also been many image processing approaches to RDT capture that do not require a physical adapter. Feng et al. [7] demonstrated that an augmented reality headset (e.g., Google Glass) can be used to track, segment, and analyze an RDT cassette that had been modified with QR fiducials for 3D tracking. Scherr et al. [24] explored the notion of using a smartphone to capture an RDT in an unconstrained manner, but their work focuses strictly on interpretation. In their study, Scherr et al. took photographs of RDTs in laboratory lighting conditions and then manually cropped those photographs before having them analyzed by custom software on a desktop. Perhaps the work most similar to our own is a commercial product called the Novarum DX Mobile Reader². Novarum's app provides real-time guidance to help people capture and interpret RDTs that are specially designed for their app; unfortunately, there have not been any publications on the usability of their app.

Despite the numerous efforts in the space of automatic RDT capture and interpretation, our work simultaneously addresses multiple gaps that previous efforts have individually left unaddressed. RDTScan is designed to work in most environments without the need for additional hardware or RDT modifications. RDTScan provides real-time guidance to ensure that high-quality photographs are taken while being mindful of computational overhead for low-end smartphones. Furthermore, we evaluate the efficacy of RDTScan in the field through quantitative analysis and qualitative feedback.

3 MOTIVATION FOR AN RDT CAPTURE APP

Smartphone adoption has been growing at a rapid pace in sub-Saharan Africa. The Global System for Mobile Communications (GSMA) reports that as of 2018, 74% of the population (774 million) has a SIM connection and 44% of the population (456 million) has a unique mobile subscription [9]. Smartphone adoption has been



Figure 3: When people are asked to take a photograph of an RDT on their own without feedback, they do not always take a photo that can be reviewed by a supervisor.

driven by the availability of affordable low- to middle-end smartphone options, particularly with the Android operating system. For example, Transsion's Tecno POP2 phone costs around \$100 USD, but only has 1 GB of RAM; in comparison, Google's Pixel 4 smartphone is \$700 USD and has 6 GB of RAM.

As smartphones become increasingly popular in sub-Saharan Africa, community health programs are starting to incorporate them into their workflows. For example, tools like Open Data Kit [4], CommCare [16], Community Health Toolkit³, and OpenSRP⁴ facilitate schedule management and data collection for patient records. The trend is no different for RDT workflows since smartphones can be used to check expiration dates and submit test results. Nevertheless, some clinics still require their CHWs to return the completed RDTs so that a supervisor can check their work. It is natural to question whether a photograph taken with the smartphone's native camera app would be a suitable substitute for the physical RDT. We asked CHWs to do this as part of our informal formative work. Even with some training, we found that many images were not suitable for review (Figure 3). Images were often out-of-focus, not at a proper distance from the RDT, or in poor lighting conditions, making it so that the important visual markers on the RDT were not clearly visible. The native camera app does not provide real-time feedback, so CHWs were not always sure that their image would be clear enough for their supervisor unless the image gets sent to their supervisor; however, that is not always feasible given battery and connectivity constraints. Introducing a physical stand to control the photography process is one way to approach this problem [5, 6] but introduces additional burden to existing practices. CHWs travel from home to home, carrying various equipment to deliver point-of-care services. Their often rugged and remote work environment brings with it a higher risk that additional hardware may break, necessitating repair or replacement. A solution for image capture without a stand would (1) remove the cost and complication of stand replacement, repair, and delivery, (2) reduce the number of items that a CHW needs to carry with them to provide care during home visits, and (3) save the time otherwise required to set up a stand at each household, which could then be devoted to other patient needs.

With this ecosystem in mind, our goal is to support CHWs in capturing high-quality RDT photographs without the need for additional hardware. As we designed our app, special attention was paid to three factors: (1) computations that could be efficiently

²<https://www.novarumdx.com/>

³<https://communityhealthtoolkit.org/>

⁴<http://smartregister.org/>

run on low- to middle-end smartphones, (2) real-time guidance for untrained CHWs, and (3) accurate representations of the RDTs.

4 RDTSCAN DESIGN

In this section, we describe how RDTScan assesses image quality in real-time for RDT capture. RDTScan is designed for Android smartphones since they are predominant in sub-Saharan Africa [9], but RDTScan could be easily ported to another operating system.

4.1 Hardware Configuration

Modern smartphone cameras have reliable auto-focus, auto-exposure, and auto-white-balance functions that automatically adjust the camera's properties with respect to whatever objects are in their field-of-view. Since Android 5.0, Google has supported the Camera2 API for explicit camera hardware control; RDTScan takes advantage of these functions to improve image quality. By default, these automatic hardware functions adjust camera parameters with respect to the whole frame. Because RDTs are narrow, relying on this default can cause the camera to optimize parameters for the background rather than the RDT itself. To yield a high-quality RDT image, RDTScan is configured so that the camera's automatic hardware functions only consider the center of the camera's field-of-view where the RDT should be located.

Since incident lighting is critical to image quality [18], RDTScan keeps the flash on continuously to provide consistent lighting underneath the smartphone. Doing so also provides more consistency with the camera's automatic hardware functions, particularly auto-exposure and auto-white-balance.

4.2 Image Quality Assurance

As the camera continuously captures new frames, RDTScan checks the RDT's brightness, sharpness, position, and size in real-time. We use different image processing techniques to check each of these aspects to either accept a high-quality image or provide feedback to the user on how to improve the image quality. Many of these checks depend on the RDT's design, so RDTScan requires an ideal, tightly cropped image of the RDT ahead of time; we call this image the *RDT template*. Below, we explain each of the image quality checks and how they are computed in real-time. Any thresholds listed in this section were empirically chosen through iterative testing.

4.2.1 Brightness. An under-exposed image makes it difficult to distinguish the RDT's test lines from the strip itself, while an over-exposed image can wash out faint test lines. Therefore, there should be a balance in how bright the RDT appears in the frame to ensure that the RDT can be properly reviewed after capture. Most RDTs are printed on white cassettes and strips, which means that little information about the RDT's brightness is lost when it is converted to grayscale. Grayscale conversion is useful because analyzing a single color channel saves computation for each incoming frame. To summarize the overall brightness of the image, RDTScan converts the image to the HLS (hue-lightness-saturation) color space and finds the maximum value of the L channel. If the highest value in the histogram is 255 (the maximum possible value), the image is likely clipped and is thus over-exposed. If the brightest value is under 125 (roughly half the maximum possible value), the image is likely too dark and is thus under-exposed.

4.2.2 Sharpness. Poor image sharpness, also known as blurriness, can happen if the camera is poorly focused or if the user is moving the camera while capturing an RDT. Blurry images have few intense edges, while sharp images have a variety of intense and non-intense edges. RDTScan uses the Laplacian operator [20] to calculate edge intensity and then examines the variance of edge intensity as a proxy for sharpness. The ideal amount of variance depends on the RDT's design, so the app computes the Laplacian variance of the RDT template to serve as a baseline. During capture, RDTScan computes the Laplacian variance for incoming camera frames specifically in the middle of the camera frame. If needed, the camera frame is scaled to the same resolution as the RDT template to ensure that the images are comparable. The camera frame passes the sharpness check if its Laplacian variance is no more than 30% lower than the reference image's variance.

4.2.3 Position, Size, and Orientation. Ensuring that RDTs are in the same position, size, and orientation across images not only guarantees that the RDT's result window will be at a reasonable viewing angle, but also facilitates quick review by supervisors. As supervisors move from image to image, they can become accustomed to where they should look for test lines and avoid wasting time searching for them. Controlling the RDT's size within the camera frame is also important for keeping the RDT outside of the camera's minimum focal distance.

Computing the RDT's position, size, and orientation requires identifying the bounding box corners of the RDT in the camera frame. RDTScan locates the RDT using feature matching between the RDT template and the camera frame. Template matching is more scalable than model-based recognition since the prior only requires a single clean image for recognition, whereas the latter requires a training dataset. Template matching using features is preferred over other criteria (e.g., cross-correlation [8]) since it is scale- and rotation-invariant, which is important for providing users with feedback. RDTScan uses the SIFT feature detector [13] to extract unique keypoints from a reference image and a camera frame. Unfortunately, running SIFT on a low-end mobile device is slow and expensive. The app mitigates this issue by downsampling the camera frame to 50% of its original size to reduce computation time. Using a brute-force matcher, the app attempts to identify a one-to-one correspondence across keypoints between the two images; the app ignores matches if the second-best candidate keypoint would produce a similar match score. A least-squares procedure is then used to calculate a 3×3 homography matrix that maps (x, y) coordinates in the RDT template to (x', y') coordinates in the camera frame. The homography matrix is used to map the corners of the RDT template to the boundary of the detected RDT. RDTScan converts the detected boundary to a rectangle if it is skewed by taking the average of opposite sides and placing them around the center of the boundary. RDTScan checks the position, size, and orientation of the rotated rectangle as follows:

- **Position:** The app measures the distance between the center of the rotated rectangle and the center of the camera frame. The frame passes this check if the distance is within 10% of the image height.



Figure 4: A screenshot of RDTScan in Bambara, a lingua franca in Mali.

- **Orientation:** The app measures the angle between the rotated rectangle's longer axis and the vertical axis of the camera frame. The frame passes this check if the angle is between -10° – 10° .
- **Size:** The app first computes the ideal size of the RDT in the camera, which we set to be the RDT template's area after scaling it to 60% of the camera frame's height. We selected 60% of the camera frame's height because it maximizes the size of the RDT in the camera frame while keeping the RDT outside of most cameras' minimal focal length. The app then measures the area of the rotated rectangle and compares it to the ideal size. The frame passes this check if the rectangle's area is within 50–70% of the image height.

4.3 User Interface

RDTScan is meant to be used without supervision, so real-time automatic guidance helps users adjust the camera as needed. The feedback is shown through our user interface (Figure 4).

4.3.1 Viewfinder. The viewfinder at the center of the interface provides a visual target where the RDT should land within the camera frame. As such, the viewfinder implicitly communicates the position, orientation, and size expectations of the image quality checks. The viewfinder has the same aspect ratio as the RDT itself, with a height that is 60% of the screen's height.

4.3.2 Real-Time Guidance. Below the viewfinder, RDTScan shows which quality checks are being failed so the user can identify concurrent issues. Real-time guidance is provided above the viewfinder to help the user remedy those issues. When the detected RDT is not in the center of the viewfinder, the app instructs the user to move the camera in one of four directions (up, down, left, right) to correct the issue. When the detected RDT is too small relative to the viewfinder, the feedback guides the user to move the camera closer to RDT; the opposite is done when the detected RDT is too

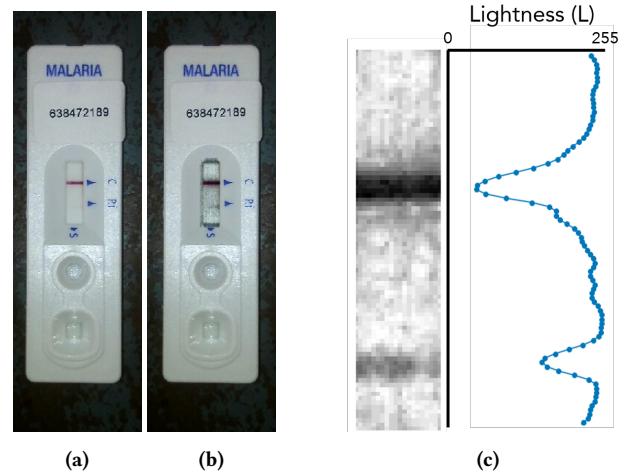


Figure 5: Examples of the SD Bioline Malaria Ag P.f cassettes that were used in our studies. (a) An example of a faint, but positive test result; (b) The image after contrast enhancement to emphasize the faint *pf* test line. (c) Our peak-finding algorithm identifies subtle but present lines in the result window.

large. Only one instruction is shown at a time to avoid information overload.

Providing real-time guidance for RDT capture is not trivial because both the user and the smartphone hardware are simultaneously trying to optimize image quality. If the user moves the camera while the hardware is trying to auto-focus, the user will disrupt the auto-focus and delay capture. Fortunately, Android provides information about the current state of the camera hardware. The video frame is likely to be blurry when the camera is either unfocused or in the middle of being focused. Therefore, RDTScan skips the image quality checks in those two states to preserve battery life and resume them once when the camera is focused.

4.4 Result Window Enhancement

Once all of the quality checks are passed, RDTScan processes the image to emphasize the details that might have been lost during capture. To do this, the app first locates the result window relative to the rest of the cassette's bounding box using its known geometry and the homography matrix calculated earlier. The app then crops and enlarges the window to make it easier for the user to see its contents. The app removes an additional 10 px border around all sides (for a 120×30 px result window) to avoid the inclusion of shadows that may appear due to the depth of the result window. To increase the contrast of faint lines, the app applies histogram equalization of the image's L channel in the HLS color space. The histogram generally skews towards the strip's bright white background; thus, global brightness equalization would darken many of those white pixels, making faint lines less distinctive. Instead, RDTScan uses contrast-limited adaptive histogram equalization (CLAHE) [21] to equalize the brightness histogram within small tiles (5×5 px). Figure 5a and 5 show a case where CLAHE reveals a line that is not clearly visible in the original image.

	Devices		
	POP2	W3	WX3
Proc. time - Setup	5.12 s	4.46 s	4.37 s
Proc. time - Exposure	15 ms	18 ms	17 ms
Proc. time - Blur	28 ms	36 ms	27 ms
Proc. time - Matching	300 ms	460 ms	310 ms
Memory Footprint	113.8 MB	104.9 MB	107.4 MB
Battery Drain Rate	846.9 mA	1033.7 mA	990.8 mA
	21.17%/hr	41.35%/hr	39.63%/hr

Table 1: A summary of system performance for three different low- to middle-end Android smartphones.

4.5 Result Interpretation

When an image is captured, the lines can be read to make a decision about the test results. The presence of a control line indicates that the test was conducted properly, and the presence of a test line indicates that the patient has malaria. The cassette can be read in two different ways: (1) supervisors or CHWs can manually interpret the results or (2) the results can be automatically interpreted with a computer vision-based algorithm. For automatic interpretation, RDTScan calculates the average lightness value in the HLS color space for each line parallel to the direction of the control line within the result window (i.e., rows in Figure 5c). When the lightness values are plotted against their corresponding positions, lines appear as troughs. To avoid false positives, the app requires that troughs be at least 3 px wide and 65 units deep ($L \in [0,255]$). The expected position and color of the lines are also used to verify that detected lines are actually part of the RDT.

5 SYSTEM EVALUATION

Before deploying RDTScan to CHWs and their supervisors, we first evaluated its performance in terms of processing time, memory footprint, and power consumption.

Processing time: For this analysis, we break down the RDTScan algorithm into two steps: hardware configuration (i.e., starting the camera and setting its parameters) and software (i.e., real-time quality checks, template matching). We measure the processing time for each step on the three devices that were used in our studies: the Tecno WX3, the Tecno W3, and the Tecno POP2. To control other factors that can affect processing time, WiFi and Bluetooth were turned off and all other apps were closed. We ran RDTScan ten times on each device with an RDT cassette in the camera frame to ensure that each quality check would be ran. Table 1 summarizes the processing time for camera setup and each of the following software procedures: exposure checking, blurriness checking, and feature matching. We found that configuring the camera hardware takes 5 seconds, which is important to consider when analyzing capture time from the user's perspective later in the paper. For the computer vision computations, feature matching took 10 times as long as the exposure and blurriness checks. In aggregate, those operations took 340–520 ms depending on the smartphone, which translates to 2–3 frames per second.

Memory usage: We also evaluated RDTScan's memory usage with Android Studio's Memory Profiler tool. Similarly to our measurements of processing time, we ran the app ten times with all

three smartphones. On average, the app used around 110 MB. This memory usage only happens during the capture process, which we limit to 30 seconds to avoid deadlocks in CHWs' workflows. Given that the smartphones we tested have 1 GB RAM and the operating system uses around 500 MB, the app's memory footprint does not put too much overhead on the device.

Battery drain: Because CHWs are out in the field when they administer RDTs, it is important to consider the power consumption that the app incurs. To measure this, we ran the app on each smartphone for an hour, forcing them to run the computer vision operations with the flash on the entire time. All other apps and features (e.g., Bluetooth, Wi-Fi, GPS) were turned off during this test, and the screen brightness was fixed to 50% to emulate screen conditions in a standard lighting environment. We analyzed the resulting battery drain rate using Android's Battery Historian.

As shown in Table 1, the Tecno POP2 lost 21% of its charge in one hour, while the Tecno W3 and WX3 lost 41% and 40% of their charge, respectively. This statistic is a bit misleading because Tecno POP2 is a newer smartphone model and thus has more battery capacity than the other smartphones (4,000 mAh vs. 2,500 mAh). The Tecno POP also consumed less current than the others, which can be attributed to improved power management in later builds of Android. Regardless, this evaluation shows that low-end smartphones with 4000 mAh battery capacity can run the app for around 6 hours continuously, which is far longer than what CHWs need for a single day.

6 STUDY 1: TECHNOLOGY EVALUATION

To evaluate RDTScan, we partnered with a non-profit organization in Mali called Muso. Among their other services, Muso employs CHWs who regularly perform malaria testing using RDTs for doorstep care to community members. Before putting RDTScan in the hands of CHWs in the field, we set out to determine whether photographs taken using our app would serve as suitable proxies for physical RDTs. To this end, we carried out a non-inferiority trial with lab technicians to confirm that readings from our app would not be unacceptably worse than direct readings by experts from physical RDTs.

6.1 Participants

Two lab technicians with expertise in malaria diagnosis were recruited from the Malaria Research Training Center (MRTC) in Bamako. The first technician was a 47-year-old male with 15 years of experience reading RDTs, while the second technician was a 37-year-old male with 10 years of experience.

6.2 Procedure and Apparatus

Before patient recruitment began, the two technicians independently practiced reading malaria RDT cassettes. The technician were guided by a third and more experienced technician to ensure systematic errors in readings were not being made. After the independent training, the technicians came together and discussed any discrepancies between their reading technique to ensure their readings were consistent.

New RDTs were activated by clinicians in Yirimadio, a peri-urban area on the outskirts of Bamako. Patients who presented symptoms

Original Image Reads	Direct Reads		Total	Enhanced Image Reads		Total	Automatic Interpretation	Direct Reads		Total
	Pos (+)	Neg (-)		Pos (+)	Neg (-)			Pos (+)	Neg (-)	
Pos (+)	100	1	101	105	2	107	Pos (+)	103	6	109
Neg (-)	7	667	674	2	666	668	Neg (-)	4	662	666
Total	107	668	775	Total	107	668	Total	107	668	775

Table 2: Confusion matrices showing how the interpretation results from (left) original image readings, (middle) enhanced image readings, and (right) automatic analysis by an algorithm compare against direct readings of the physical RDT cassettes.

of malaria or any signs of severe illness that would normally be tested for malaria had their blood drawn and applied to RDTs. After the 20-minute wait-time required for RDT activation, one of the technicians recorded their *direct reads* of the RDT cassette in a spreadsheet. Within 15–30 minutes, a research staff member took a picture of the RDT using the RDT capture app installed on one of two Tecno WX3 smartphones. The WX3 has a 5-inch screen, 854×480 pixel display, and 5 MP camera. The smartphones were used in such a way that each lab technician had a smartphone that stored images of the RDTs they interpreted. The research staff ensured that there were no external cues present in the image that could make the RDT uniquely identifiable or reveal its test result to the technicians. Image capture was conducted in a room with similar lighting conditions as the rooms where the lab technicians were reading the physical RDT cassettes.

Each lab technician interpreted the images to produce an *original image read* for each RDT. The images were shuffled to avoid ordering effects and ensure that the sequence did not make specific images recognizable. Two months later, when the enhancement algorithm was developed, the images were post-processed, re-shuffled, and then read again by the same people to produce corresponding *enhanced image reads*.

6.3 Analysis

6.3.1 Interpretation by Experts. We assessed RDTScan’s performance by measuring the concurrence between direct reads of the RDT cassettes and the various interpretation mechanisms enabled by RDTScan: original image reads, enhanced image reads, and automatic interpretation. We calculated concurrence using sensitivity (true positive rate) and specificity (true negative rate) along with their 95% confidence intervals (CIs). Direct reads were treated as the ground truth since that is the current best practice for many organizations in low- and middle-income settings. RDTs were excluded from our analysis if the test results were inconclusive (e.g., no control line, excessive smudging).

A total of 795 images were captured at the clinic, of which 107 were positive, 668 were negative, and 20 were indeterminate according to direct reading; therefore, our analysis was conducted on 775 images. The underlying prevalence of malaria in our study population was 13.8%.

The left side of Table 2 compares original image reads and direct RDT reads. Interpreting the unmodified images taken by RDTScan resulted in a sensitivity of 93.5% (CI: [87.0%, 97.33%]) and a specificity of 99.9% (CI: [99.2%, 100%]). The higher specificity over sensitivity can be attributed to the fact that some of the positive RDTs had faint lines that were even more difficult to read when they were

digitized. The middle of Table 2 shows the concurrence between the enhanced image reads and the direct reads. This led to a sensitivity of 98.1% (CI: [93.4%, 99.8%]) and a specificity of 99.7% (CI: [98.9%, 100%]). The sensitivity improved by roughly 5% because contrast enhancement made faint lines more obvious and thus reduced false negatives. The false positives resulted from image noise that formed at the test line location after contrast enhancement. The false negatives were due to lines that were faint even after contrast enhancement.

6.3.2 Interpretation by an Algorithm. The right side of Table 2 compares the performance of automatic interpretation against direct reads. Our algorithm was comparable to expert reads, achieving 96.3% (CI: [90.7% to 99.0%]) sensitivity and 99.1% (CI: [98.1% to 99.7%]) specificity. Compared to the enhanced image reads, there were two more false negatives and four more false positives, which could be attributed to two factors: (1) noise generated after the contrast enhancement and (2) blood within the result window. Blood can create false positives or false negatives depending on how it stains the RDT strip. If the stain is throughout the strip, any lines on the strip are rendered indistinguishable from the background; if the stain is localized near the test line, the blood can mimic a positive strip result. Both cases occurred in our dataset, generating a few more false negatives and positives than enhanced image reads.

7 STUDY 2: DEPLOYMENT STUDY WITH CHWS

Having demonstrated strong concurrence between direct reads of RDT cassettes and images from RDTScan, we deployed our app to CHWs in the field. The purpose of this deployment was to assess RDTScan’s usability and efficacy in real-world environments and to collect feedback for future app iterations.

7.1 Participants

We recruited supervisors and CHWs from two program sites in Yirimadio and Tori. Two supervisors and six of their subordinate CHWs were randomly selected from each clinic. The 12 CHWs who participated in our study (1 male, 11 females) were between 23–53 years old and had varying levels of experience reading RDTs (2 months–6 years). These CHWs and their supervisors use smartphone apps built using the Community Health Toolkit⁵ in their routine activities, so they were deemed to have a relatively high degree of readiness to adopt new smartphone apps for their work.

⁵<https://communityhealthtoolkit.org>

7.2 Procedure and Apparatus

The deployment study was run for ten weeks between May–August 2019. Before the study started, a half-day training session was held to introduce CHWs to our RDT capture app. The session covered instructions on how to use the app, tips for optimal capture, and study logistics. We had an additional training session for the supervisors on how to best support the CHWs and report app issues to the research team. Each CHW was handed a smartphone pre-installed with RDTScan. Six Tecno POP2 (5.45-inch screen, 960×480 px display, 5 MP camera) and ten Tecno W3 (5-inch screen, 854×480 px display, 5 MP camera) smartphones were used in the study. CHWs were asked to capture every RDT they conducted in the field for the entire study duration. CHWs ran a different number of RDTs and thus produced different numbers of captured images. The images were automatically saved in the smartphone with metadata on the CHW's identity, their smartphone model, a timestamp, and the time taken for RDT capture. To ensure that RDTScan did not significantly disrupt the CHWs' workflow, we implemented a 30-second timeout within the app. CHWs who were able to get a successful automatic capture in that time were taken to a screen that showed the enhanced result window alongside the original image. If an RDT was not successfully recognized before the timeout, an image of whatever was currently in the camera's view was captured. After many discussions with the clinics, we decided to not reveal the automatic interpretation results to the CHWs since doing so could have significantly altered diagnostic outcomes while the app was still under development.

7.3 Analysis

For the deployment study, we focused on analyzing the usability of RDTScan both quantitatively and qualitatively. The main quantitative measures were capture time and automatic capture rate. Two researchers reviewed the captured images to assess whether the images had sufficient quality for supervisors to review the RDTs' test results. We also conducted semi-structured interviews to get qualitative feedback from CHWs on RDTScan's usability.

We collected 533 images over the course of the study, with each user capturing 44 images on average (min: 3, max: 79). A subset of images were excluded from our analyses either because the CHWs did not follow our instructions for how to administer the RDT or the RDTs were not correctly captured by RDTScan. That subset includes 145 images where CHWs wrote on the cassettes, 11 images with RDTs that were upside down, 13 images with a smudged or covered camera, and 3 images with contaminated RDTs. We note that RDTScan is robust to writing on the cassette as long as it does not overlap with the RDT's most prominent visual keypoints, but RDTs with writing were excluded unilaterally for consistency. Therefore, we analyzed 361 images in total.

7.3.1 Overall Capture Performance. To assess RDTScan's ease of use, we calculated the average capture time and proportion of images captured before the 30-second timeout. Table 3 shows the overall capture performance in the deployment study. On average, CHWs took around 20 seconds to capture an RDT image. Automatic capture was triggered 67% of the time before the timeout, and the average capture time was 14.4 seconds in those cases. Our system performance evaluation revealed that the smartphones took

	Capture Time		Automatic Capture Rate
	w/ Timeout	w/o Timeout	
All Trials	20.3 s	14.3 s	67%
Trial 1–10	21.8 s	13.7 s	58%
Trial 11–20	19.8 s	14.9 s	72%
Trial 21–30	18.3 s	14.9 s	78%

Table 3: The average capture time and automatic capture rate across all CHWs during the deployment.

5 seconds on average to configure the camera's hardware, thus accounting for one-quarter of the average capture time. In other words, CHWs only spent 15 seconds interacting with the app once the camera was ready. Within that time, CHWs usually spent 3–5 seconds positioning the camera relative to the RDT, and the remaining 10 seconds was spent waiting for a frame that would pass all of the quality checks.

7.3.2 Image Quality. All captured images were assessed by two researchers along three criteria: size, position, and brightness. Any disagreements were resolved after by the researchers coming together and discussing the relevant images. In the end, we found that all 354 captured images clearly showed the result window, which is the most critical piece of information during review. However, there were 27 images that did not clearly show the entire RDT; 4 images were over-exposed, 13 images were taken too close, 10 images were taken too far, and 3 images were off-center. Although CHWs exceeded the timeout 33% of the time, the images that were captured at the end of that period typically passed our image quality criteria. We suspect that although the template-matching algorithm may have failed to identify the RDT, the viewfinder interface and real-time guidance helped CHWs position the RDT properly relative to the camera.

7.3.3 Capture Performance Improvement. The deployment study lasted for over 3 months. Since the CHWs were new RDTScan users, we were interested in investigating how quickly they were able to learn how to use the app. We first compared the automatic capture rate and average capture time during their 1st–10th, 11th–21st, and 21st–30th trials. As the bottom two rows in Table 3 show, the automatic capture rate improved from 58% in the first 10 trials to 72% to 78% in second and third set of 10 trials, respectively. We found that the average capture time for automatic capture stayed the same, suggesting that the CHWs were able to capture RDTs fairly consistently.

We further analyzed how capture success rate changed over time. Figure 6 shows this trend across all CHWs. The graph shows a steady improvement in automatic capture rate as the CHWs gained more experience with RDTScan. After 25 captures, CHWs were successful more than 80% of the time; after 30 trials, that number rose further to 93%. We further analyzed whether there were differences between different CHWs. For each CHW, we compared their automatic capture rate between their first 10 trials and trials 20–30. One CHW was excluded from this analysis because they only used the app three times. Figure 7 summarizes the capture rate for the eight remaining CHWs. Five of the CHWs improved over time, with two people reaching 100% by their 20th trial. Unfortunately, CHWs who

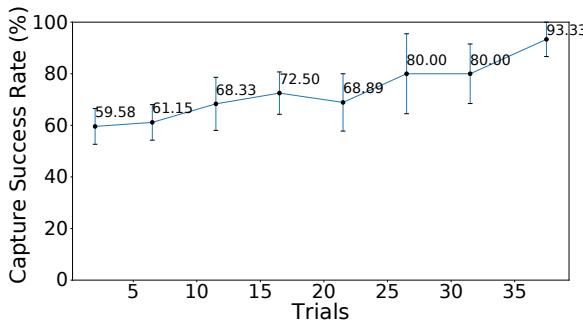


Figure 6: The change in automatic capture success rate over time. The error bars show standard error.

started with poor capture rates were more likely to not improve. In our study, supervisors did not provide additional training or support apart from issues like app crashes; in practice, supervisors could provide struggling CHWs with additional guidance to avoid repeated issues.

7.3.4 Qualitative Feedback. We conducted semi-structured interviews with both the CHWs and their supervisors. For the CHWs, the questions were mainly around the usability and utility of RDTScan, along with any difficulties they experienced during the study. For the supervisors, the questions involved the utility of the app in their RDT workflow. The findings are summarized below.

Improved workflow: The overarching objective of RDTScan is to improve the RDT workflow of CHWs in the field. All of the CHWs agreed that RDTScan made their jobs easier since it voided the need for them to physically return RDT cassettes to clinics for review. Two CHWs pointed out that RDTScan enabled instant feedback from their supervisors when images could be uploaded, which helped them make a prompt decision on whether to treat the patient or readminister the RDT.

Although the CHW supervisors were not direct RDTScan users, the app still brought value to their workflow. One supervisor mentioned that the app eliminated the work he had to do to collect, store, and dispose of all the RDTs returned by his CHWs. The app also reduced the uncertainty that was caused when supervisors received RDTs that may have deteriorated over time. With RDTScan, the images effectively freeze the state of the RDT in time to make it as if the supervisor is with the CHW when they conduct their review. When images can be uploaded immediately after an RDT is administered, supervisors said they could provide quick feedback that would help CHWs decide whether to prescribe an anti-malarial, readminister the RDT, or clear the patient.

Accurately representing RDTs: Aligning with the findings of our first study, all of the CHWs agreed that the captured images were accurate representations of the RDT cassettes. The supervisors also believed there was no substantial difference between the images and the actual RDTs, with one person saying, “[I] have confidence in the RDT image tool as the images are clear and the result readable.”

Difficulties: We asked about any difficulties or issues using RDTScan. Most of the CHWs had little experience taking photos with a smartphone, even for their personal use, so the experience of using RDTScan was completely new to them. One CHW pointed out

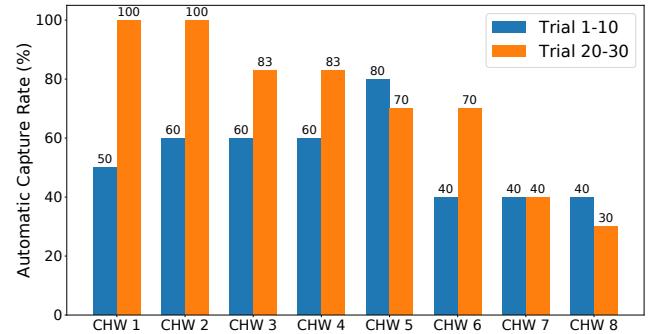


Figure 7: The change in automatic capture success rate over time separated by CHWs. Most of the CHWs were more successful using RDTScan over time, with some exceptions being the CHWs who had the least success early on.

that it was difficult for her to follow the instructions while trying to get all the quality checks passed when she first used the app, but she had more success as she became more experienced with the app. This remark aligns with the previous quantitative analysis that shows improved automatic capture rates over time. Other CHWs commented that the battery drained quickly over the course of their workday. We believe that the novelty of camera-based smartphones played a small role in this, leading CHWs to underestimate the amount of battery life required to use the smartphone’s camera and flash multiple times a day. Nevertheless, battery usage is still a major concern that cannot be ignored when designing for CHW workflows. Most of the images after the 30 second timeout were high-quality, thanks in part to the app’s instructions and interface, so we hypothesize that a shorter timeout could avoid long capture times without sacrificing image quality.

8 DISCUSSION

The goal of our work was to simplify CHWs’ RDT workflows by taking advantage of smartphones’ hardware and portability. We demonstrated that CHWs and their supervisors were able to use RDTScan to capture images that accurately represent RDT results. We also showed that CHWs were able to use RDTScan with reasonable success in the field. Below, we describe the implications and limitations of our work.

8.1 Implications of Automatic Interpretation

Our conversations with stakeholders in RDT workflows have revealed that some people are hesitant to deploy a tool that automatically interprets RDTs on the CHWs’ behalf. Although most people believe that automatic interpretation could reduce errors and give CHWs more confidence in their own decisions when they agreed with the algorithm, they also worry about the consequences of when CHWs disagree with the algorithm, regardless of whether the latter is right or wrong. In our study, we decided to show CHWs the enhanced RDT image to *support* their existing decision-making processes rather than possibly *overriding* them.

Beyond evaluating the automatic interpretation algorithm in more diverse scenarios, we foresee numerous opportunities to make it more acceptable in different contexts. Comparing the intensity

of the test line against the control line could generate a confidence value that CHWs could use to rule out weak decisions or to highlight images that warrant manual inspection by CHW supervisors. The algorithm could consider the local prevalence of malaria to apply a Bayesian perspective to screening. RDTScan could even enable a new workflow where CHWs upload images to supervisors who run the interpretation algorithm to aid their decision-making before giving instructions to the CHWs.

8.2 Limitations of Smartphones

Smartphones are preferred over other devices like laptops and tablets because of their portability and ability to serve multiple purposes (e.g., phone calls, SMS messaging). Even so, smartphones are not without their limitations. When one lab technician re-visited images of positive RDTs that he had initially determined were negative, he said, “It is clear that the images are positive on the computer screen, but on the smartphone screen it is sometimes hard to be sure because the line is so faint”. The preference of viewing images on a laptop instead of smartphones was echoed by other individuals, primarily because the laptop’s larger screen size (15 inches vs. 5.5 inches) allowed them to make the images much larger and thus easier to read. Color response, brightness, and resolution can also vary between different screens and impact how people interpret RDT images. If global health organizations are interested in deploying tools like our RDT capture app for on-site image interpretation, they may want to consider the affordances of the devices that display those images. From our experiences, we believe that tablets may provide an optimal balance between cost, portability, and screen size to promote accurate image interpretation.

Limited image storage is also a barrier for RDTScan deployments. For our study, images were saved locally on smartphones and uploaded to cloud storage once network connectivity was satisfactory. Low-end smartphones often have limited storage space, and the cost of data uploading is non-trivial in low-income settings. In the future, we believe that data compression could be used to mitigate some of these costs.

8.3 RDT Co-Design for Image Processing

We designed RDTScan to accommodate a variety of RDT designs. However, our feature-matching approach is not a perfect solution. Feature-matching requires the existence of unique visual keypoints that span the result window. Such keypoints usually exist due to the natural presence of labels, arrows, and branding images, but some RDT designs lack enough features for robust matching. Any markings over visual keypoints on the RDT can also complicate detection.

There are other RDT characteristics that are suboptimal for image processing. Many cassettes have a deep window showing the underlying test strip, which can create shadows at the edges of the result window. Some RDT designs also have a thin film that protects the test strip but creates a reflection under certain lighting conditions. We plan on working with RDT manufacturers to create an RDT that is optimized for image processing. The ideal feature list includes a wide and flat cassette, non-reflective materials, and explicit fiducials for recognition.

8.4 Expediting the Review Process

Although RDTScan is meant to alleviate the need for CHWs to transport RDTs back to their supervisors in the clinics, supervisors still wanted to review the decisions that the CHWs made. This review process can be burdensome across an entire organization when each supervisor has around 10–20 CHWs for whom they are responsible. Clever grouping and sorting of the RDT photographs could expedite this process in the future. When RDTs are grouped by a particular characteristic, supervisors can assess a representative subset of images within each group. For example, if a supervisor examines a subset of images captured by a particular CHW and concurs with all of their decisions, the supervisor may accept all of the other decisions made by that CHW and turn their attention to one with more disagreements. Our image processing pipeline can also contribute to an intelligent ordering within these groups, sorting images according to line intensity and allowing supervisors to prioritize faint lines over intense or non-existent ones.

9 CONCLUSION

In this work, we proposed a smartphone-based RDT capture app that generates an accurate image representation of RDTs. RDTScan provides real-time feedback for capturing high-quality images in a way that is designed for low- to middle-end smartphones. We showed that the captured images were accurate in representing the actual RDTs, achieving 98.1% sensitivity and 99.7% specificity compared to readings from actual RDTs. We also demonstrated that CHWs were able to use RDTScan with great success, capturing a high-quality image within 18 seconds on average at a rate of 78% after 20 trials. We hope that this work inspires other camera-based health applications designed for low-resource settings in the future.

10 ACKNOWLEDGMENTS

Funding for this study was provided by the Bill and Melinda Gates Foundation (Grant #OPP1184006, OPP1183755). We thank Arunan Skandarajah, Libby Lavitt, and Richard Anderson for providing valuable feedback on this work. We also thank all of the researchers, CHWs, and supervisors for their assistance in this work: Jenny Liu for sample size calculations for the sensitivity and specificity protocol, Mahamadou Sogoba for data entry during our technology evaluation, Kassoum Kayentao for intellectual review, Caroline Whidden for intellectual review and CHWs supervision, and Mariam Cisse and Ibrahim Cisse for enabling the Yirimadio CSCOM as a test site.

REFERENCES

- [1] Chukwuemeka CA Azikiwe, C C Ifezulike, Iyeopu M Siminalayi, Louis U Amazu, J C Enye, and O E Nwakwunite. 2012. A comparative laboratory diagnosis of malaria: Microscopy versus rapid diagnostic test kits. *Asian Pacific Journal of Tropical Biomedicine* 2, 4 (apr 2012), 307–310. [https://doi.org/10.1016/S2221-1691\(12\)60029-X](https://doi.org/10.1016/S2221-1691(12)60029-X)
- [2] Lacey Best-Rowden, Shiwani Bisht, Joshua C. Klontz, and Anil K. Jain. 2014. Unconstrained face recognition: Establishing baseline human performance via crowdsourcing. In *IJCB 2014 - 2014 IEEE/IAPR International Joint Conference on Biometrics*. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/BTAS.2014.6996296>
- [3] Ross M Boyce, Anthony Muiru, Raquel Reyes, Moses Ntaro, Edgar Mulogo, Michael Matte, and Mark J Siedner. 2015. Impact of rapid diagnostic tests for the diagnosis and treatment of malaria at a peripheral health facility in Western Uganda: An interrupted time series analysis. *Malaria Journal* 14, 1 (may 2015), 203. <https://doi.org/10.1186/s12936-015-0725-0>

- [4] Waylon Brunette, Samuel Sudar, Mitchell Sundt, Clarice Larson, Jeffrey Beorse, and Richard Anderson. 2017. Open Data Kit 2.0. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services - MobiSys '17*. ACM Press, New York, New York, USA, 440–452. <https://doi.org/10.1145/3081333.3081365>
- [5] Nicola Dell and Gaetano Borriello. 2013. Mobile tools for point-of-care diagnostics in the developing world. In *Proc. DEV '13*. ACM Press, New York, New York, USA, 1. <https://doi.org/10.1145/2442882.2442894>
- [6] Nicola Dell, Ian Francis, Haynes Sheppard, Raiva Simbi, and Gaetano Borriello. 2014. Field evaluation of a camera-based mobile health system in low-resource settings. In *Proc. MobileHCI '14*. 33–42. <https://doi.org/10.1145/2628363.2628366>
- [7] Steve Feng, Romain Caire, Bingen Cortazar, Mehmet Turan, Andrew Wong, and Aydogan Ozcan. 2014. Immunochromatographic diagnostic test analysis using google glass. *ACS Nano* 8, 3 (mar 2014), 3069–3079. <https://doi.org/10.1021/nm500614k>
- [8] A Goshitasby, S. H. Gage, and J. F. Bartholic. 1984. A Two-Stage Cross Correlation Approach to Template Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6*, 3 (1984), 374–378. <https://doi.org/10.1109/TPAMI.1984.4767532>
- [9] GSMA. 2019. *The Mobile Economy Sub-Saharan Africa*. Technical Report. www.gsmaintelligence.com
- [10] Steven A Harvey, Larissa Jennings, Masela Chinyama, Fred Masaninka, Kurt Mulholland, and David R Bell. 2008. Improving community health worker use of malaria rapid diagnostic tests in Zambia: package instructions, job aid and job aids-plus-training. *Malaria Journal* 7, 1 (dec 2008), 160. <https://doi.org/10.1186/1475-2875-7-160>
- [11] Sócrates Herrera, Andrés F Vallejo, Juan P Quintero, Myriam Arévalo-Herrera, Marcela Cancino, and Santiago Ferro. 2014. Field evaluation of an automated RDT reader and data management device for Plasmodium falciparum/Plasmodium vivax malaria in endemic areas of Colombia. *Malaria Journal* 13, 1 (mar 2014), 87. <https://doi.org/10.1186/1475-2875-13-87>
- [12] Alinune N Kabaghe, Benjamin J Visser, Rene Spijker, Kamija S Phiri, Martin P Grobusch, and Michèle Van Vugt. 2016. Health workers' compliance to rapid diagnostic tests (RDTs) to guide malaria treatment: A systematic review and meta-Analysis. *Malaria Journal* 15, 1 (dec 2016), 163. <https://doi.org/10.1186/s12936-016-1218-5>
- [13] David G Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 2 (2004), 91–110. <https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf>
- [14] Christine Manyando, Eric M Njunju, Justin Chileshe, Seter Sizya, and Clive Shiff. 2014. Rapid diagnostic tests for malaria and health workers' adherence to test results at health facilities in Zambia. *Malaria Journal* 13, 1 (may 2014), 166. <https://doi.org/10.1186/1475-2875-13-166>
- [15] Jonathan Mayes and Andrew White. 2016. How Smartphone Technology Is Changing Healthcare In Developing Countries. *Journal of Global Health* (2016).
- [16] Gayo Mhila, Brian DeRenzi, Caroline Mushi, Timothy Wakabi, Matt Steele, Prabhjot Dhaldialla, Drew Roos, Clayton Sims, Jonathan Jackson, and Neal Lesh. 2009. Using Mobile Applications for Community-based Social Support for Chronic Patients. *Health Informatics in Africa* (2009). [//publication/uuid/04129DC2-6E77-4B4B-A1AE-B0402F593EA4](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.187.6186&rep=rep1&type=pdfhttp://www.cs.washington.edu/homes/bderenzi/Papers/mhila_jhelina09.pdf%5Cnpapers2:)
- [17] Onur Mudanyali, Stoyan Dimitrov, Uzair Sikora, Swati Padmanabhan, Isa Navruz, and Aydogan Ozcan. 2012. Integrated rapid-diagnostic-test reader platform on a cellphone. *Lab on a Chip* 12, 15 (aug 2012), 2678–2686. <https://doi.org/10.1039/c2lc40235a>
- [18] National Instruments. 2017. A Practical Guide to Machine Vision Lighting - National Instruments. (2017). [http://www.ni.com/white-paper/6901/en/#toc5](http://www.ni.com/white-paper/6901/en/http://www.ni.com/white-paper/6901/en/#toc5)
- [19] Haydar Ozkan and Osman Semih Kayhan. 2016. A Novel Automatic Rapid Diagnostic Test Reader Platform. *Computational and Mathematical Methods in Medicine* 2016 (apr 2016), 1–10. <https://doi.org/10.1155/2016/7498217>
- [20] J L Pech-Pacheco, G Crist, J Chamorro-Mart Nez, and & J Fern Andez-Valdivia. [n.d.]. *Diatom autofocusing in brightfield microscopy: a comparative study*. Technical Report. www.iv.optica.csic.es/projectssautof.html
- [21] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. 1987. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing* 39, 3 (1987), 355–368. [https://doi.org/10.1016/S0734-189X\(87\)80186-X](https://doi.org/10.1016/S0734-189X(87)80186-X)
- [22] Stephen Poyer, Tanya Shewchuk, Sarah Tougher, Yazoume Ye, The Actwatch Group, Andrea G Mann, Barbara A Willey, Rebecca Thomson, John H Amasi, Ruilin Ren, Marilyn Wamukoya, Mark Taylor, Samuel Blay Nguah, Blessing Mberu, Admirabilis Kalolella, Elizabeth Juma, Charles Festo, Boniface Johanes, Graciela Diap, Katia Bruxvoort, Daniel Ansong, Kara Hanson, Fred Arnold, and Catherine Goodman. 2015. Availability and price of malaria rapid diagnostic tests in the public and private health sectors in 2011: results from 10 nationally representative cross-sectional retail surveys. (2015). <https://doi.org/10.1111/tmi.12491>
- [23] Esmée Ruizendaal, Susan Dierickx, Koen Peeters Grietens, Henk DFH Schallig, Franco Pagnoni, and Petra F Mens. 2014. Success or failure of critical steps in community case management of malaria with rapid diagnostic tests: A systematic review. , 229 pages. <https://doi.org/10.1186/1475-2875-13-229>
- [24] Thomas F Scherr, Sparsh Gupta, David W Wright, and Frederick R Haselton. 2016. Mobile phone imaging and cloud-based analysis for standardized malaria detection and reporting. *Scientific Reports* 6, 1 (sep 2016), 28645. <https://doi.org/10.1038/srep28645>
- [25] Osama M.E. Seidahmed, Muneir M.N. Mohamedinein, Afrah A. Elsir, Fayez T. Ali, El Fatih M. Malik, and Eldirdieri S. Ahmed. 2008. End-user errors in applying two malaria rapid diagnostic tests in a remote area of Sudan. *Tropical Medicine & International Health* 13, 3 (feb 2008), 406–409. <https://doi.org/10.1111/j.1365-3156.2008.02015.x>
- [26] Severe Malaria Observatory (SMO). 2019. *Mali / Severe Malaria Observatory*. Technical Report. <https://www.severemalaria.org/countries/mali>
- [27] Chansuda Wongsrichanalai, Mazie J Barcus, Sinuon Muth, Awalludin Sutamihardja, and Walther H Wernsdorfer. 2018. A Review of Malaria Diagnostic Tools: Microscopy and Rapid Diagnostic Test (RDT). *The American Journal of Tropical Medicine and Hygiene* 77, 6_Suppl (2018), 119–127. <https://doi.org/10.4269/ajtmh.2007.77.119>
- [28] World Health Organization and Center for Disease Control. 2010. *Basic malaria microscopy*. 1–88 pages. <https://books.google.com/books?hl=en&lr=&qid=Uo59VLcUmRMC&oi=fnd&pg=PP2&dq=giemsa+malaria&ots=gF6PB2a9gd&sig=2d6p5B5hpOPHNL-wH3Eqyqdt0RU>