Topographical Computer Vision Algorithms for rapid, low-cost Hematological diagnostics and Parasite detection through Random Forests classification and van Leeuwenhoek-type Imaging

Tanay Tandon
Cupertino High School

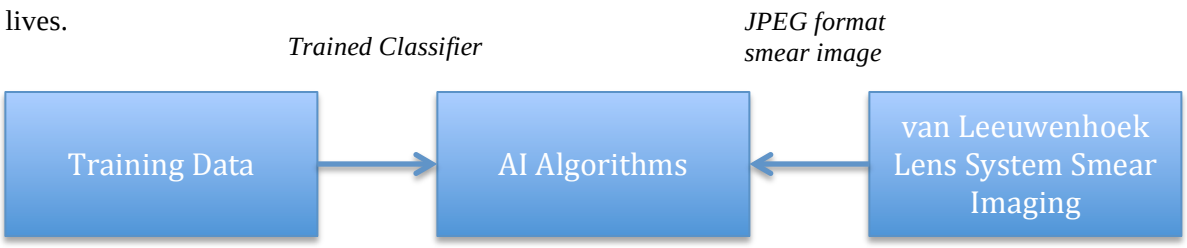
I. Abstract

This interdisciplinary study develops a low-cost, portable device for automated blood diagnostics and parasite detection through a novel machine-learning based computer vision algorithm for blood cell analysis on smartphone CMOS (camera) systems. Rural regions lack access to expensive in-lab diagnostic equipment and trained pathologists for disease detection through smear analysis. Thus, millions of potentially treatable cases go undiagnosed, leading to high parasite mortality rates in said areas. The developed Topographical Euclidean distance transformation approach through multivariate local maxima peak analysis and PCA derived Random Forest (RF) classification automatically identifies parasites in blood smears with a 0.73-0.85 accuracy comparable to human gold-standard of 0.9. The algorithm utilizes Otsu clustering for thresholding and segmentation, using a labeled dataset of 1248 smear cells to train the RF ensemble in a projected 10-dimensional feature space. An engineered van Leeuwenhoek type lens system attachable to a smartphone camera interfaces with this computer vision approach to image blood samples with 360x magnification and 480uM field of view, algorithmically classifying and counting cells in the sample. Overall, the developed computer vision model and low-cost lens imaging system provide a rapid, portable, and automated blood morphology test for rural region disease detection, where in-lab equipment and trained pathologists are not readily available.

I. Introduction

More than 1.5 million people in rural regions die every year due to undiagnosed, yet highly treatable parasitical infections[6]. This is mainly due to lack of access to lab-centric setups and the necessity of a skilled microscopist for visually diagnosing conditions such as Malaria, Chagas, and Toxoplasmosis. Thus, there is need for a portable, low-cost, and automated blood test easily available in all corners of rural regions for rapid analysis of cell morphology. Such a device would assist in saving millions with undiagnosed cases by algorithmically identifying parasite presence, and analyzing hundreds of cell samples without the presence of a human pathologist.

This study proposes a novel machine learning based model for automated blood diagnostics using an engineered portable van Leeuwenhoek lens system attached to a cellphone CMOS sensor. The developed Topographical Euclidean distance transformation algorithm through multivariate local maxima peak analysis and PCA-derived Random Forest classification automatically analyzes the visual features of blood cells to detect parasite presence and produce cell counts. The algorithm was developed in the Python language, using collected training data to teach the Random Forest ensemble by distinguishing features in cell types. The van Leeuwenhoek system utilizes a ball lens to optically magnify blood samples (collected from Carolina Biological Supply), and image on the smartphone for analysis. Testing was conducted by cross validating model performance against a human gold standard of labeled data. By means of this Artificial Intelligence approach and lens system, the algorithm automatically identifies diseases within the blood stream on a portable device, independent of lab environments or a morphologist, leading to a cheap smear analysis system for rural regions, potentially saving thousands of lives.



Collected CDC smear data and cell imaging data is morphologically labeled and used to grow Random Forest trees in ensemble. PCA projects high dimensional feature vectors to 10 dimensional space for training, generating the trained Forest.

Smear image is **segmented** into constituent cell parts by means of Topographical representation. Spliced portions are **classified** using trained Random Forest and PCA.

Van Leeuwenhoek type lens system engineered by identifying focal region of 1mm lens, and deployed onto CMOS sensor through additive manufactured frustum.

II. Materials and Methodology

Experimentation was conducted independently using CDC (Center for Disease Control) public smear imaging data combined with Peripheral Blood Sample Set 312041A acquired from the Carolina Biological Supply. The development and research occurred primarily in the Python programming language on the Sci-kits learning platform (scientific computation library), Matplotlib, Numpy, and OpenCV imaging libraries. The paper will first discuss the development of the **computer vision models (Section 2.1),** and then transition to the engineering of the **van Leeuwenhoek type smartphone lens imaging system for blood cell magnification and analysis (Section 2.2).** The interfacing of the two is discussed in the final section **(Section 2.3), where full blood tests for parasite detection and cell counts are conducted on the phone system and backend.**

2.1 Computer Vision – Topographical Segmentation and Random Forest Classification

Random Forests (RF) are statistical learning ensembles originally proposed in Briemann (2001) [1] to optimize on information gain for regression or classification tasks, by creating a series of weak learners (trees) selected to contribute to an eventual strong learner (the forest). Traditionally, RFs have found usage in text-classification tasks, sentiment analysis, and topic segmentation.

This study proposes and develops a stochastic Random Forest ensemble approach for cell-type classification from labeled training data. Sample predictions are made by averaging the regression for an observation 'X', throughout the weak learning trees – f_b as shown in equation 1, where B is typically a free parameter.

Eq.1 Random Forest Prediction Aggregating (Bagging)

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} \hat{f}_b(x')$$

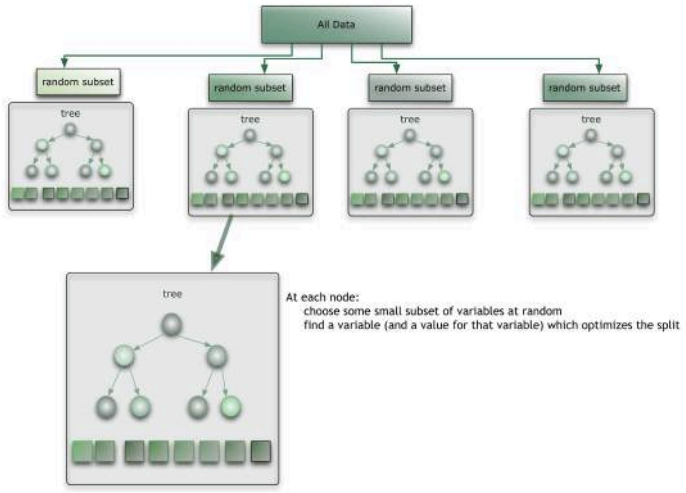


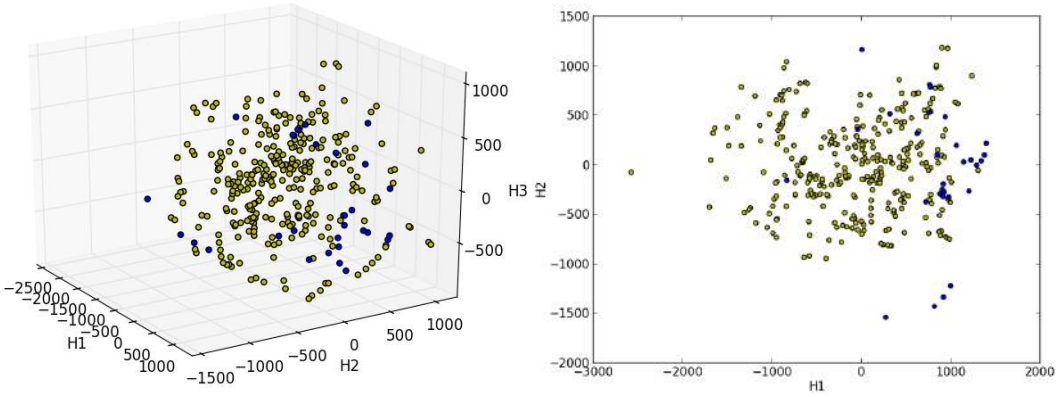
Fig. 1 (right) Random Forest Overview [1]

The stochastic nature of Random Forests causes randomized subsets of training data to be coupled with randomized feature nodes, thereby reducing effects of overfitting on datasets.

The developed Random Forest module was trained with 10-dimensional data, projected from the high dimensional feature space of the original image to a lower dimensionality through Principal Component Analysis (PCA). A Sci-kits decomposition Randomized PCA function was implemented to conduct dimensionality reduction on the given cell data types (discussed further in later sections).

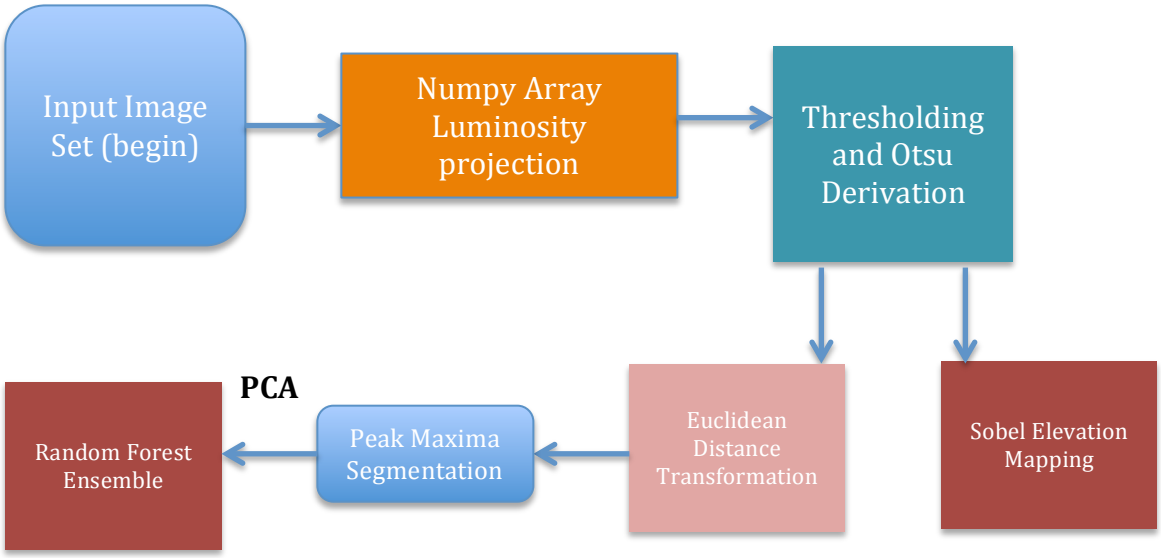
Fig. 2 Randomized PCA

A 3 dimensional feature space rendering of smear pixel data via PCA. This is also projected to a secondary 2 dimensional space, through the same PCA approach. Each scatter-point represents a single cellular artifact, with yellow representing RBC, and blue indicating Leukocyte. The Axes are PCA defined components of the n-dimensional feature vector.



The task of training and developing the models can be observed by the following bird's eye view:

Fig. 3 Computer Vision Model Overview



Essentially, two major sections exist in this model – **Segmentation** and **Classification.** Segmentation takes in the raw captured smear and identifies single cellular units (referred to as 'artifacts' henceforth) in the smear by means of Luminosity projection, Otsu thresholding, Euclidean Transformation, elevation mapping, and local maxima analysis. Classification is conducted by growing the Random Forests Ensemble with the dimensionally-reduced data from the PCA function of the segmented cell types

(labeled). The grown, trained ensemble can then classify unseen segmented data following the training procedure. The development and theory behind segmentation and classification is explained in detail within the following sections.

2.1.1Segmentation 2.1.2 Classification

2.1.1 Segmentation – Otsu Thresholding, Euclidean Distance Transformation, Peak Maxima Analysis

2.1.1.a Luminosity Projection

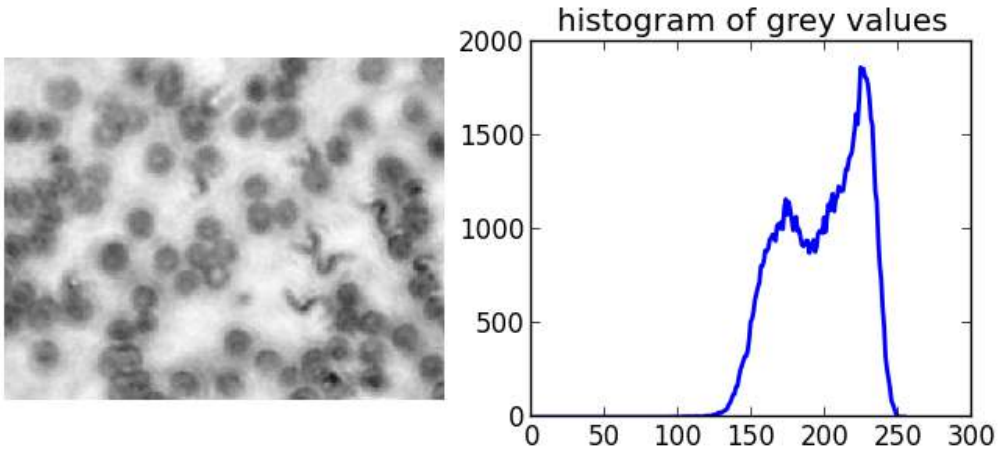
The first stage developed in segmentation is the projection of the RBG image to a numpy intensity map (entirely grayscale). The Luminosity approach is taken to generate the intensity map as follows:

Eq. 3 Luminosity Projection

$$L(rgb) = \{0.21\ R + 0.72\ G + 0.07\ B\}$$

Weighting the colors accordingly to take into account human perception (higher sensitivity to green). A generated intensity map of an example Trypanosoma Smear collected from the lens system (lens system engineering discussed in section 2.2) is shown below. The intensity histogram is shown beside it.

Fig. 4 Intensity map generation and Histogram

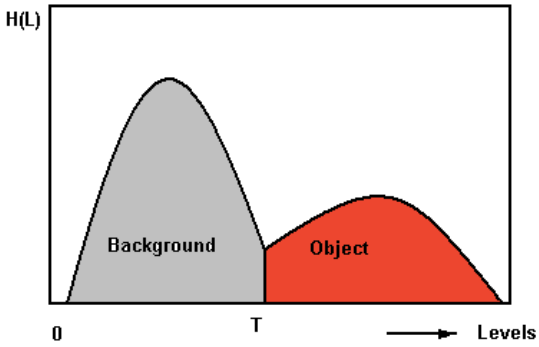


Peaks are shown accordingly in the foreground and background. Trypanosoma Parasites are clearly visible in the smear image (left). The Histogram: (y axis frequency, x-axis intensity value)

The Luminosity function generates the grayscale image, along with the accompanying histogram indicating a clear foreground and background (indicated by the spikes in the graph). Following the intensity mapping, there is still observable noise surrounding the image cells. Thresholding can algorithmically identify and truncate this noise, leaving only artifacts of interest (foreground), for continued cellular analysis. This process is most easily paralleled to the subconscious procedure a human pathologist would conduct in distinguishing cells from background.

Fig. 5 below shows the generalized developed algorithm for Otsu histogram-based thresholding:

2.1.1.b Otsu Thresholding



$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)\left[\mu_1(t) - \mu_2(t)\right]^2$$

Fig. 5 Bi-modal Histogram Thresholding (left) Eq. 4 (right) Otsu's method for threshold identification

Otsu (above) clusters the binary classes (foreground and background) by minimizing intra-class variance, which is shown to be the same as maximizing inter-class variance[4]. Thus, the optimal image threshold can be found, and, the intensity map may undergo foreground extraction. A Trypanosoma smear sample is once again used, below to highlight the Otsu function of thresholding:

Fig. 6: Otsu derived Threshold for smear image

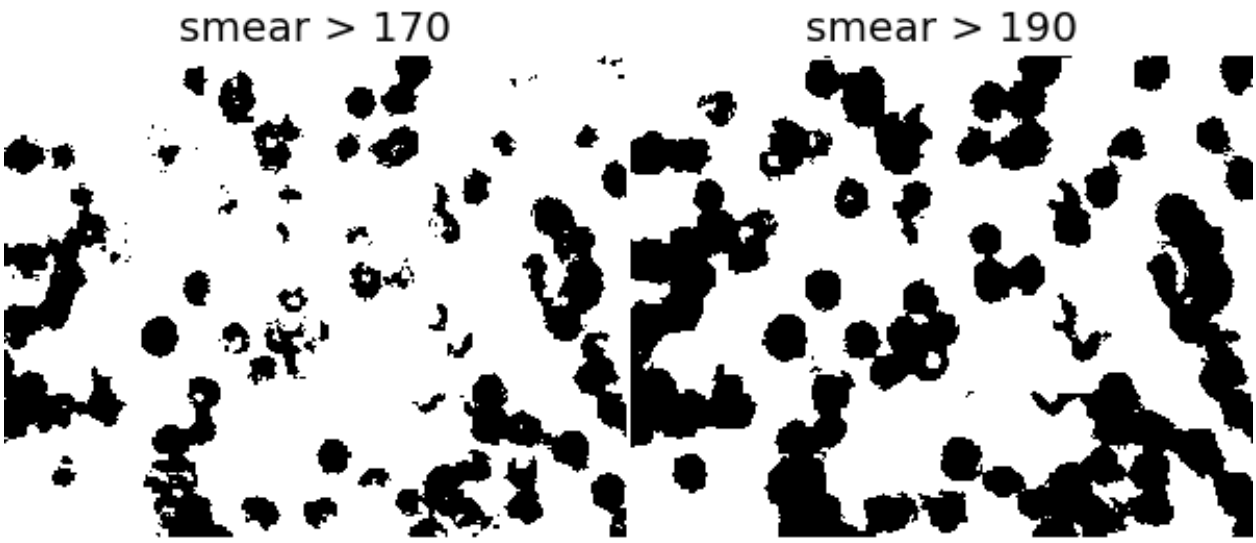


Fig. 6 above highlights the Otsu threshold transformation, with 190 being the calculated optimum threshold for the particular smear. A lower threshold is provided to highlight less than optimal threshold value for binary foreground classification.

Foreground-Background engineering being complete, the next stage focuses on transforming the peripheral smear shot to segment into constituent artifacts for training and classification. The process can be most clearly paralleled to the procedure undertaken by a pathologist in analyzing each cell independently, differentiating it from surrounding cells. A distance transformation topographically defines the image in context of boundary pixels. The purpose of the application is to identify 'peaks' and 'valleys' in the graph in classifying a cell cluster, thus distinguishing from a surrounding body.

2.1.1.c Topographical Euclidean Distance Transformation

Eq. 5 Euclidean Distance Generalized Equation

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2+(q_2-p_2)^2+\cdots+(q_n-p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

$$\|\mathbf{p}\| = \sqrt{p_1^2+p_2^2+\cdots+p_n^2} = \sqrt{\mathbf{p}\cdot\mathbf{p}}$$

The distance to the nearest boundary (foreground-background boundary) is calculated by means of the E. Dist. Function, and is derived for the intensity value of the given pixel region.

The Distance Transformation utilizes the threshold intensity map to define the topographical region of the cells in the image. As shown above, the Euclidean algorithm re-formats the intensity plot based on the boundary location, hence successfully defining the intensity of each pixel in the artifact as a function of its enclosure by the blob. This transformation allows for blob analysis by defining object peaks, or locations most consumed by the surrounding pixels as local maxima. The Euclid space simply reformats the two dimensional pixel array in accordance to distance between pixel_n and the background-foreground boundary - it does not actually splice the two into segmented images. As shown in Fig. 7-10 below, the intensity topography generated by the Euclid function can then be plotted in a three-dimensional space to characterize cell boundaries, and identify regions of segmentation and body centers.

Fig. 7

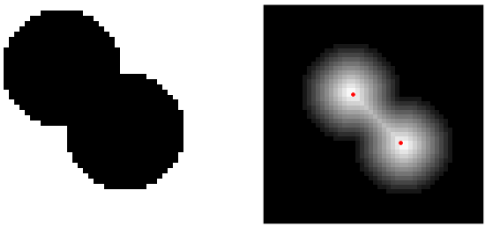


Fig. 8

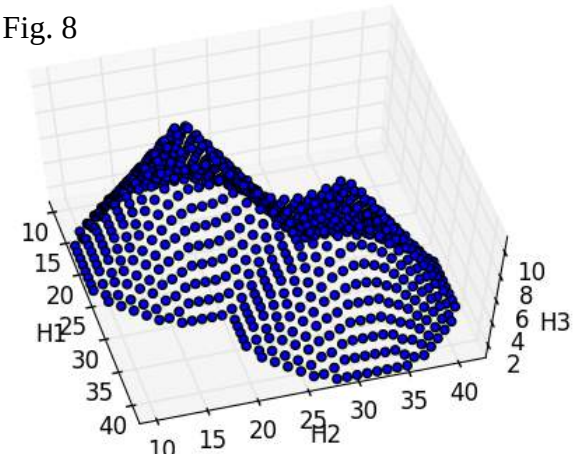


Fig. 9

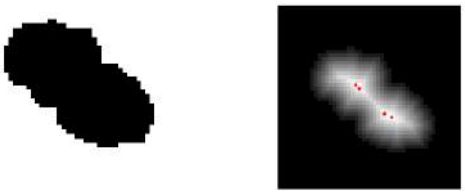


Fig. 10

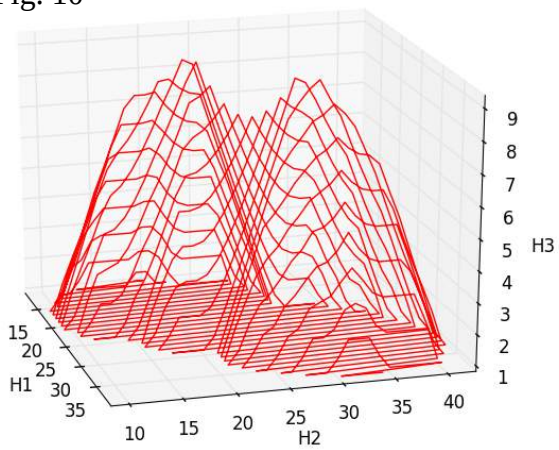


Fig. 7 above, Simulation Cell Euclidean Distance Transformation: The Euclidean transformation is conducted on a simulated cell body and the corresponding Surface Intensity Plot (Fig. 8) is shown. Fig. 9 indicates the same procedure on a RBC cell sample with the intensity plot shown as a wireframed Surface plot on the right (Fig. 10). The red points in Fig. 7 and Fig. 9 are calculated local_maxima based on multivariate numerical calculus analysis. Explained in detail below.

0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0

⇒

0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0
0	1	2	2	2	2	1	0
0	1	2	3	3	2	1	0
0	1	2	2	2	2	1	0
0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0

Fig. 11 Right – Simple matrix Euclidean distance transformation for n dimensional space.

Using this expressed definition of Euclidean Transformation for intensity mapping, we can see above a simple matrix transformation example. Below (fig 12, 13) is the Euclidean Transformation conducted on the original threshold data from smear image. The threshold smear is now newly mapped through this transformation, and the generated numpy array can be passed to multivariate maxima identification.

Fig. 12

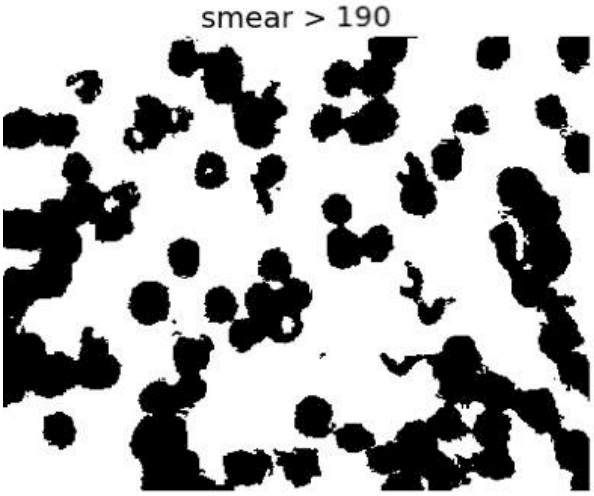
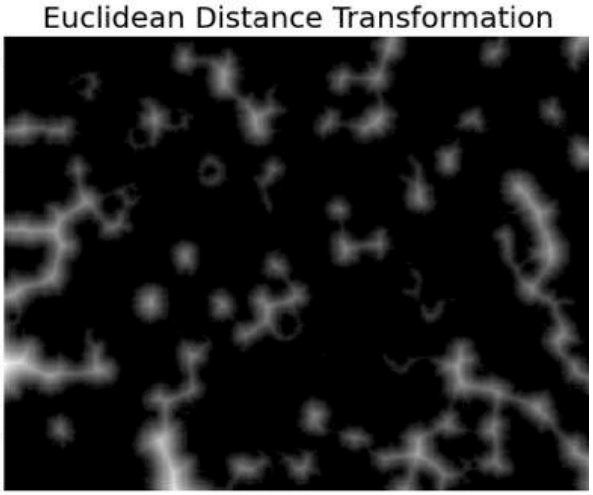


Fig. 13



2.1.1.d Local Maxima Peak Analysis

It is assumed that the local maxima (see fig 8, 10) in the Euclidean Transformed image will indicate artifact body centers, leading to a location for segmentation. The local maxima are derived through numerical calculus, by finding peaks in the 2 dimensional numpy arrays (peaks highlighted with red points Fig. 14). These peaks are used as the coordinates for segmentation, and define splicing rectangles for extracting cell bodies from the smear shot.

Fig. 14

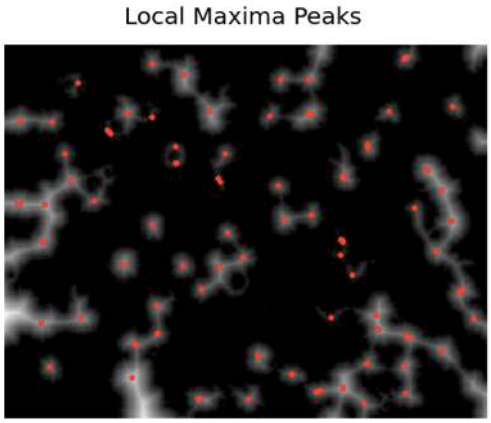
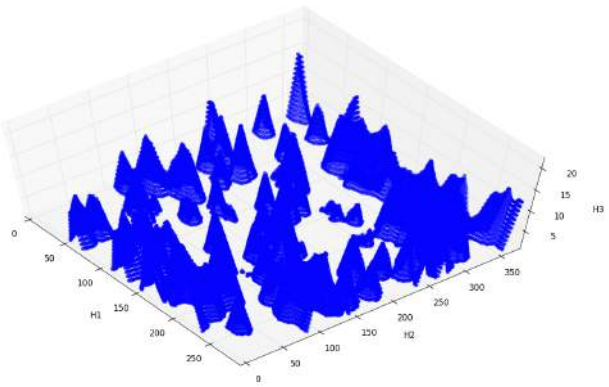


Fig. 15 Full Smear maxima Surface Plot



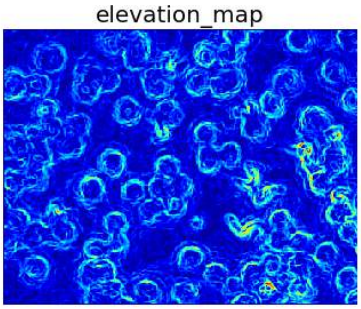
Given the Euclidean peak region identifications, the image is then spliced based on the derived artifact dimensions from Sobel filtering elevation map generation (below). The Elevation map algorithm uses a Sobel operator to approximate the size of each artifact and conduct the cell extraction accordingly.

Eq. 6

$$\mathbf{G}_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A}$$

Above, the Sobel Operator (Sobel Filter) is used to recreate processing image A with highlighted prominence on edges. This creates an elevation map of the image that is combined with the Euclidean Transformation to segment and splice the cells. To the right, the generated elevation map for a blood smear with the Sobel edges highlighted.

Fig. 16



The spliced image is generated through the *numpy sub_array function* passing the Euclidean and Sobel rectangular values as parameters, resulting in a dataset of segmented cells from the original smear image shot. The coordinates extracted from the Euclidean distance transformation and local maxima calculation are applied back to the original colored image to make the rectangular segmentation of cells on the original. The cells are then normalized to a 50x50 jpeg shot for identification, Random Forest training, PCA, and algorithm accuracy cross validation. It should be noted that this segmentation procedure is independent of any later classification – regardless of cell type or morphology, the splicing procedure will extract constituent cellular artifacts. Thus, this process automates the procedure taken by a trained pathologist when examining a smear field to distinguish between cells, and initiates the process for the actual morphology and cell identification. With this splicing into constituent elements, the segmentation

portion of the paper is complete and the machine learning classification model training begins after Fig. 17 (below).

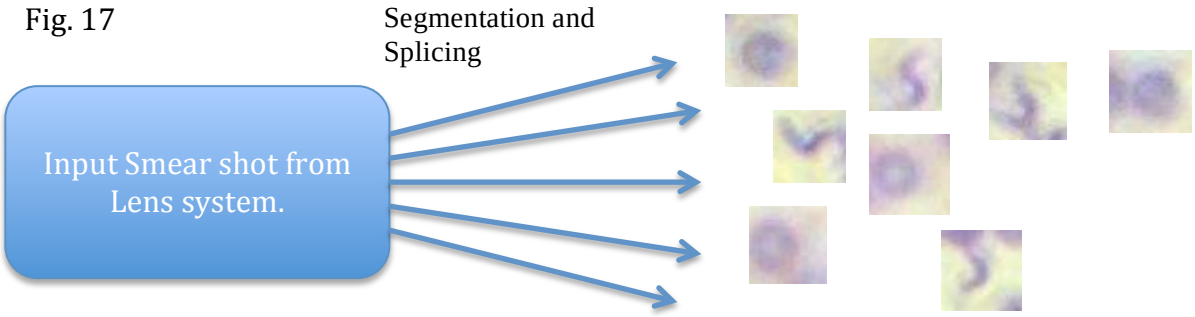


Fig. 17 above. The segmented cellular artifacts are generated by using the generated Euclidean transformation to mimic the map on the original input image and generate the separate segment images.

2.1.2 Classification – Dimensionality Projection, Random Forest Training

2.1.2.a Training Data

The spliced images are then used for training and cross validation of cell identification/classification accuracy. In cases of training data, the data is collected from CDC public smear sets and microscopy images and then labeled morphologically to establish a human gold standard. The human gold standard training labeling procedure was as follows:

Training Data Collection

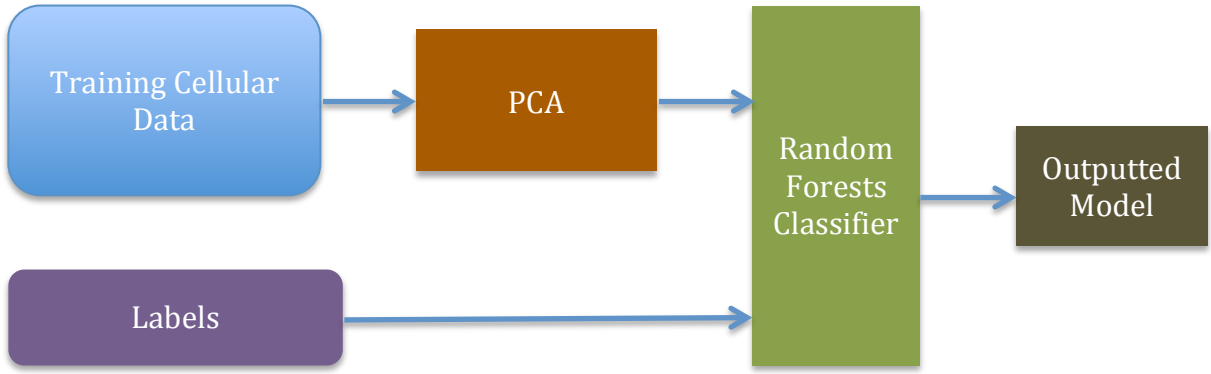
1. Collect all samples of a given parasite from CDC public repository or imaged microscopy with appropriate label assigned. Pre-labeled images are archived into directory.
2. Smear images not pre-labeled by training set or source are hand labeled in accordance with CDC morphology guidelines for specific cellular artifact type. Human hand labeling is the current gold-standard for the range of parasites worked with.
3. Any cell-type whose class label is still unclear (even after applying CDC morphology guidelines in hand-labeling) is set aside and sent to expert morphologist (see acknowledgements Dr. Niaz Banaei Stanford Infectious Disease) for secondary opinion and final labeling. This accounted for less than 10 cell samples through the course of the entire research.
4. Archive samples in Morphology_Train directory for tree-growing and training process to begin the Random Forest classification tree building process.

The final training set produced through this methodology from the CDC data and microscope imaging of Carolina Research smear samples consisted of the following types:

Sample Type	**N_Samples**
Trypanosoma Cruzi	253
Trypanosoma Brucei	248
Drepanocytosis	228
Healthy Whole Blood	282
P. Falciparum	237

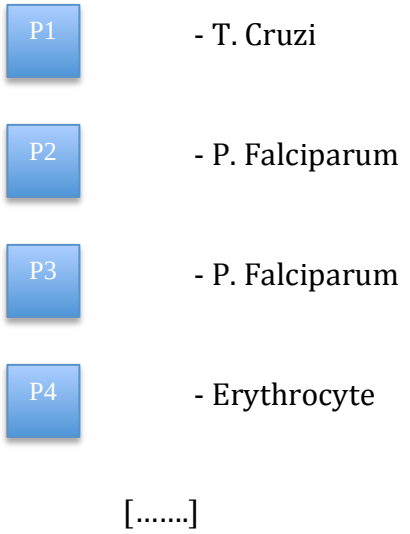
Using labeled versions of these images, the classifier was trained to identify Trypanosoma, Drepanocytosis (Sickle Cell), Plasmodium, healthy erythrocytes (Red Blood Cells), and leukocytes (White Blood Cells). The training procedure is explained in detail below:

Fig. 18



Given the stochastic nature of the designed Random Forest, data is bootstrap aggregated (Bagged), the 10 dimensional feature space generated by the RandomizedPCA algorithm is fed to the trees with randomized subsets and feature nodes. Each blood cell image (labeled in the training data) is used to seed the forest based on its assigned class label, as shown below (Fig. 19,20). The training data was hosted on a Linux box and compiled as part of the backend setup. Training procedures were run on the Linux box with the necessary computer vision scripts running remotely to optimize training times and data allocation to the RF module.

Fig. 19, 20



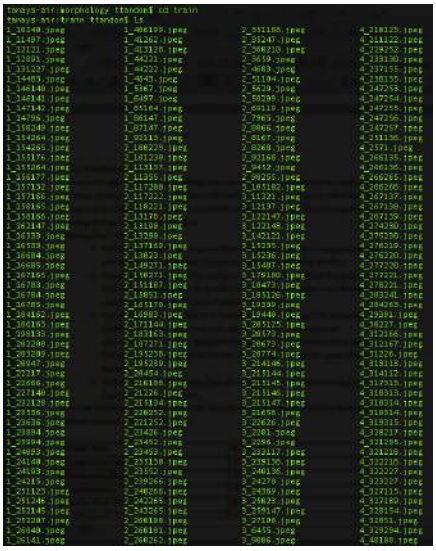


Fig. 19, 20 – The training directory structure, with images and assigned labels (classes) for each cell image in training data. On the right is a dump of the training directory with image jpeg shots used for training. The purpose of the model training is to extrapolate trends between the raw pixel data of the images and the label they are assigned. These trends are then applied to unseen images from the segmentation to tag and identify cell type. Thus allowing for cell counts and parasite disease identification.

2.1.2.b PCA Dimensionality Projections

RandomizedPCA generates a 10 dimensional feature vector from each image in the training set. Every element in the training set is represented by this multi-dimensional vector, and fed into the RF module to correlate between the label and the features. By regressing between these feature vectors and the assigned cell-type class label, the model attempts to identify trends in the pixel data and the cell type. RF selects for trees optimizing on Information Gain in terms of accuracy in predicting cell type label. Thus, after being trained, given a unseen segmented image sample, the model predicts the cell type – using the classification model to identify parasite presence and cell count based on the training set.

```
##third round of PCA with 10 components, loaded into Random Forests for classification
pca = RandomizedPCA(n_components=10)
data_points = pca.fit_transform(train_data)
print data_points

Forest = RandomForestClassifier(n_estimators = 200)
Forest = Forest.fit(data_points, label_array)

test = []

##single test case for demo
img = img_to_matrix("full_images/trypan_test1.jpg")
img = flatten_image(img)
##to allow for PCA, we must observe this image in context of the previous training data to run PCA on.
train_data.append(img)
pca = RandomizedPCA(n_components=10)
data_points = pca.fit_transform(train_data)
test_data = data_points[len(data_points)-1]
test.append(test_data)

prediction = Forest.predict(test)
print prediction
```

Fig. 21 left
Code snippet of high-level RandomizedPCA initializing, Forest initializing, training (fitting), and predicting on unseen test data sample (trypan_test1.jpg). The test data PCA is done in context of the training, to determine the eigenvector projections.

In model training, the raw pixels become features in analysis, however, given the incredibly high dimensionality of this data, it is unfeasible to train an entire classifier. The data segments – being normalized to 50 by 50 jpeg images – would each contain 2500 features leading to model over fitting probability, and unworkable training times. Hence, the dimensionality reduction through PCA for lower-dimensional data representations is conducted.

Fig. 22

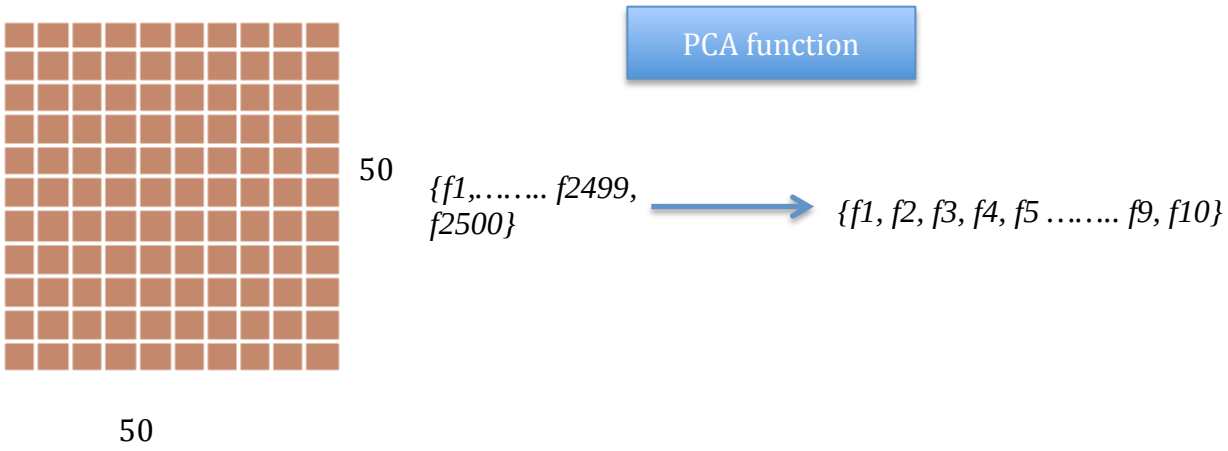


Fig. 22 above. The 2500 featured data is projected to the 10 dimensional space for faster training and better modeling (less overfitting). See Curse of Dimensionality Appendix.

2.1.2.c Random Forest Ensemble Training

The full model training procedure with PCA analysis and Random Forest data fitting requires between 30 minutes to 1 hour based on the size of the training set. The outputted classifier (Forest) is then serialized, saving the grown tree states as *.pickle* for later classification and analysis. The training procedure is only conducted once per domain set, and is then scalable to all test data within the same domain. Predictions are outputted per image in CSV structure with the image ID and class label attached. See introduction to the Computer Visions section for an overview of Random Forest processing. In fig. 21; 200 estimators (trees) are grown with the data. Each estimator receives a randomized subset of the original data and splits of at randomized feature nodes to make a prediction. The stochastic nature of the forest tends to prevent overfitting on low-dimensional datasets, however the high dimensionality of image-based spaces exponentially increases training times[1]. Thus, the lower dimensionality of the PCA application set alleviates this issue, and lowers training times. ANNs (Artificial Neural Networks) are another option to train on the raw pixel data given their greater usage in image processing, however training times and GPU resources were important factors to take into account given the on-field applications of the research.

The definition of training procedure and model-outputting marks the end of the Computer Vision algorithm section of the paper. As a summary – the paper develops a novel Random Forest based Euclidean distance transformation and local_maxima approach for automatic morphological blood cell analysis. By developing the multivariate peak analysis and distance transformation algorithms, the smear shot from the lens system can be segmented into constituent cell artifacts. These artifacts are then analyzed for pixel-label correlations to extrapolate a trained model capable of predicting morphological characteristics and cell-type. As a whole, the process mimics blood microscopy procedures conducted in-lab using human gold standards and identifies the presence of healthy erythrocytes and parasite cells within a blood smear.

2.2 Lens System Engineering, Model loading, System fabrication

Given the creation of the cell classification algorithms, the secondary study focus was the engineering of a van Leeuwenhoek type lens magnifier capable of mounting on a CMOS camera system to magnify and image blood cells. The purely automatic/algorithmic nature of the developed cell morphology process yielded the opportunity of leveraging a digital camera magnifier attached to a microprocessor for an entirely portable, independent blood diagnostic system. Most modern-day cellphones serve as both powerful processors and relatively high powered cameras, thus a smartphone-based microscope lens system was pursued in development. This section will go into detail regarding the optical design theories, system development, and model fabrication of the hardware CMOS van Leeuwenhoek based lens magnification system for low-cost blood cell analysis. The following section will then discuss the interfacing of the two, for the final blood testing system.

2.2.1 van Leeuwenhoek lens system design

Blood smear imaging occurs between the 100-1000x magnification range, with a varying range of resolutions and field sizes based on microscope model. Compound, two-stage microscopes are most commonly used in morphological examinations of blood smears for parasite diagnosis, and earlier, cell count reports. The viability of low-cost spherical glass lenses for microbiological magnification has been proposed, most notably by Antonie van Leeuwenhoek in the 17^{th} and 18^{th} centuries.

This study engineered a lens-system for CMOS sensors based on van Leeuwenhoek type designs for imaging and computer vision analysis to produce morphology reports. The low-cost nature of ball lens magnifiers, along with the mountable structure of the lens made it a viable choice for magnification production and projection back to the smartphone camera system.

Fig. 23

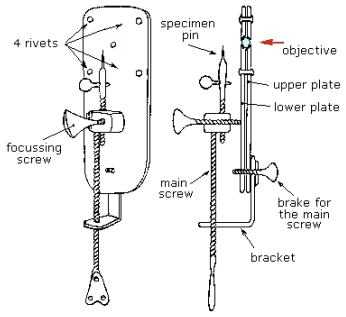
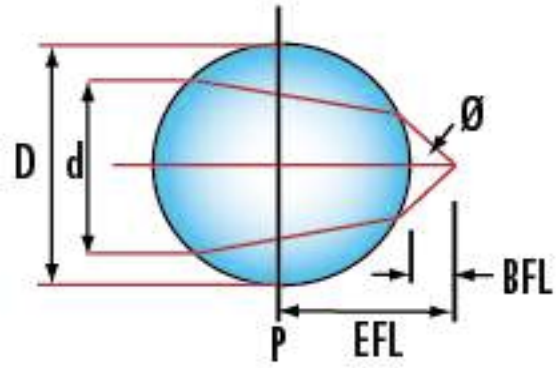


Figure 1 - Diagram of the microscope constructed by Antoni van Leeuwenhoek in the XVII century

Fig 23 Right, van Leeuwenhoek system with lens placed within screw-based brass structure. The original models made use of a spheroid lens upon which a viewer would focus the sample by turning the screws. Although a single lensed system, the van Leeuwenhoek design far outperformed other microscopes of the time at 200-300x magnification compared to 20-50x. The study mimicked a van Leeuwenhoek type setup by employing a ball 1mm ball lens capable of approaching 400x magnifications. Older van Leeuwenhoek designs required the system to be held directly to the viewer's eye. The developed design leverages a CMOS camera to ease in processing and viewing.

Fig. 24 Ball Lens Magnifier



Eq. 7, 8

$$EFL = \frac{nD}{4(n-1)}$$

$$BFL = EFL - \frac{D}{2}$$

The magnification of traditional ball-lens systems is defined as shown in Fig. 23 and equations 7 and 8 above. For this particular study, Edmund Scientific N-BK7 Ball Lenses were used in order to render magnification. A variety of lens sizes were tested with the Effective Focal Length (EFL), Back Focal Length (BFL), and Index of Refraction determined as shown above. Given the necessity of a roughly 300-400x magnification based on the prepared training data, the N-BK7 1mm lens type was determined to be optimal for camera mounting and magnification. Given the close proximity of the focal point to the lens surface, the 1mm spherical lens type required the sample to be held roughly directly in contact with mounting platform.

A 6th Generation iPod Touch was used (6 megapixel CMOS camera) in testing, coupled with a dark rubber piecing pierced to hold the lens snugly. The rubber piecing was exactly 1.1 mm thick, allowing for the placement of the lens in contact with the phone camera surface.

Fig. 25

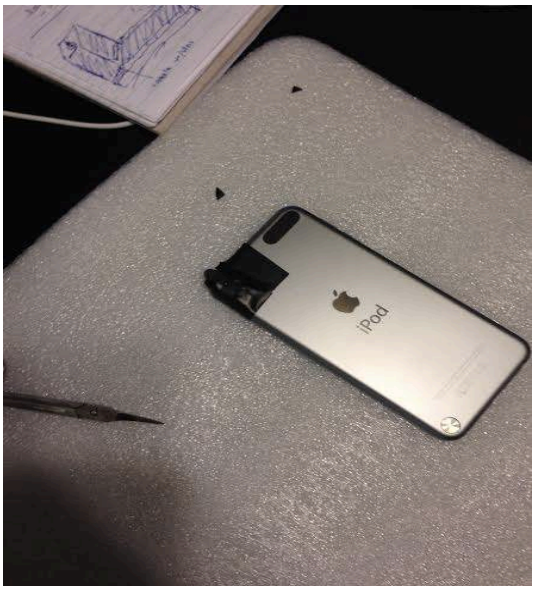
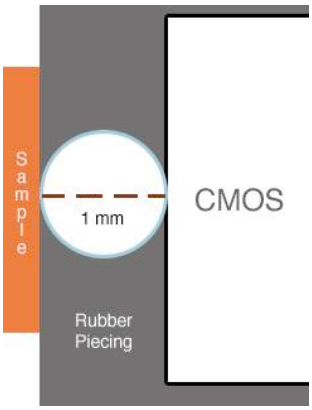


Fig 25 (left) The ball lens system planted in the rubber piecing attached atop the CMOS sensor to magnify the images.
Fig 26 (bottom right) rendering of lens setup atop CMOS sensor.
1 mm ball lens encased in rubber piece and then placed atop camera sensor. Sample is magnified by placing directly atop rubber piecing, and viewing camera reading.

Fig. 26



The original version of the lens magnifier was assembled by planting a ball lens magnifier onto the rubber piecing using bamboo optical equipment tweezers holdings. The rubber piecing was then firmly secured onto the camera surface by means of adhesive glue. Lighting was provided by means of a backlight placed behind sample, or directly underneath sampling setup. In early designs, a flash of a secondary device was used to light the sample. Fig. 27, 28 below highlights the original setup with iPod camera attachment, sample, and backlight.

Fig. 27

Fig. 28

Fig. 27, A secondary device's flash is used as a backlight tunneled through the makeshift roll. The backlight portion itself is not essential to the system, only a means to illuminate the sample (held atop). The experimenter holds the van Leeuwenhoek ball-lens system above the sample for microscopy. Fig. 28 The slide and sample shown above the backlight (without lens system). The low-cost, portable nature of the setup makes it ideal for rural region blood analysis.

This first version of the system using the rubber piecing for mounting, produced magnifications ranging between 300-360x on resolution targets with a field of view of roughly 480uM. However, due to lack of rigidity or structure around the magnifier, pollutants and particulate matter easily collected above the magnifying region. Furthermore, the ball lens was prone to movement within the structure, causing focal length changes and lens aberrations. Thus, a secondary model was developed using additive manufacturing to mount the lens in a frustum design.

Fig. 29

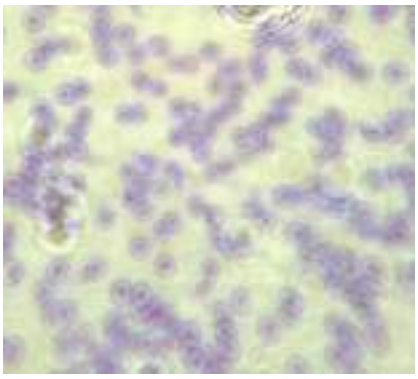
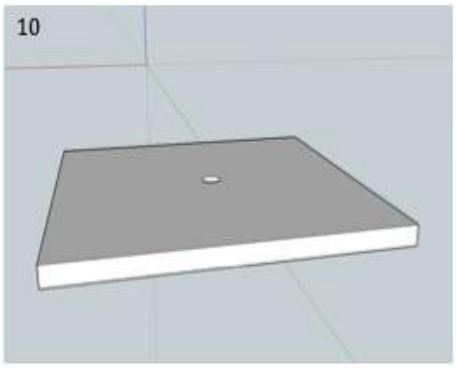
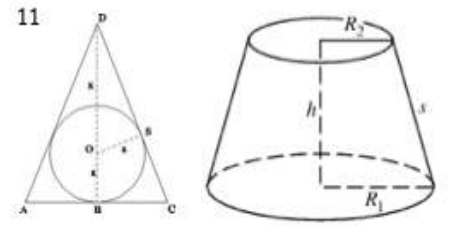


Fig. 29 left, a erythrocyte sample viewed under the engineered van Leeuwenhoek system by a iPod Touch 6th generation camera. The cells are clearly visible, but due to the lack of rigidity the ball lens has likely moved from the target region atop the CMOS aperture, moving the focus point in reference to the sample and leading to less than optimal magnification and resolution.

2.2.2 Additive Fabrication of CMOS system

To develop a solution for the issue of lens movement and particulate matter buildup, a system using additive fabrication (3d-prining) was developed. A frustum shape (see below) using PLA material was designed and printed to more securely hold the van Leeuwenhoek type system in place. The spherical lens





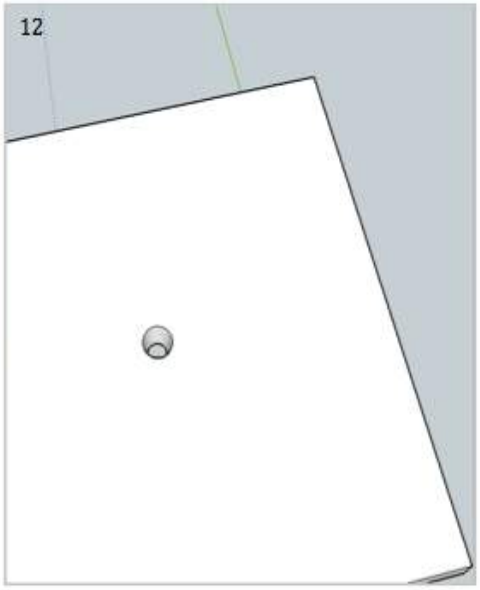


Fig 30 (top left) *A wafer upon which the frustum shape has been cut to allow lens to be placed easily inside. The larger opening allows for placement, and the gradually decreasing diameter ensures the lens does not simply fall out.* ***Fig 31 (bottom left)*** *Drawing of Frustum shape with r1 and r2 radiuses. Also shown is illustration of inscribed circle in triangle, which was used to calculate frustum dimensions.* ***Fig 32 (right)*** *A top view of the frustum opening.*

would be encompassed by the shaped wedge edges in the wafer (1.2mm opening, .8mm closing), to allow for mounting on the larger opening, and then securing in the 1mm middle diameter. Bamboo tweezers were used to mount the lens itself into the frustum, and then pressure was applied for 20 minutes to secure the lens. The resulting setup held the lens in place with a much more rigid structure and far less regions for particulate matter infiltration.

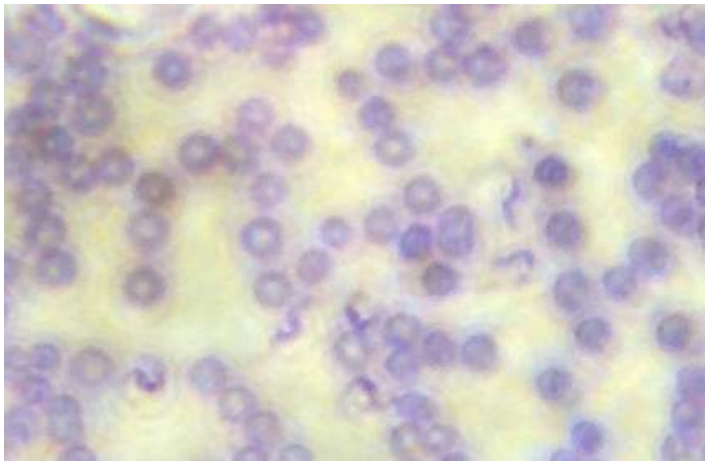


Fig. 33 left, Additively manufactured frustum wafer attached to iPod camera for magnification. Fig. 34 top, Trypanosoma smear under new lens setup. Trypanosoma parasite clearly visible in contrast to erythrocytes.

The reprinted system consistently produced lens magnifications in the 360x magnification range with a 480uM field of view, and much fewer visible particulate pollutants.

2.3 Vision and Lens System Integration

The interfacing of the trained Topographical algorithm with the engineered van Leeuwenhoek lens system mentioned in section 2.2 consisted of deploying the algorithm on a Linux box backend and building an application running on the smartphone device to send images to the server, and receive diagnostic responses.

Fig. 35-38 Classified smears on device

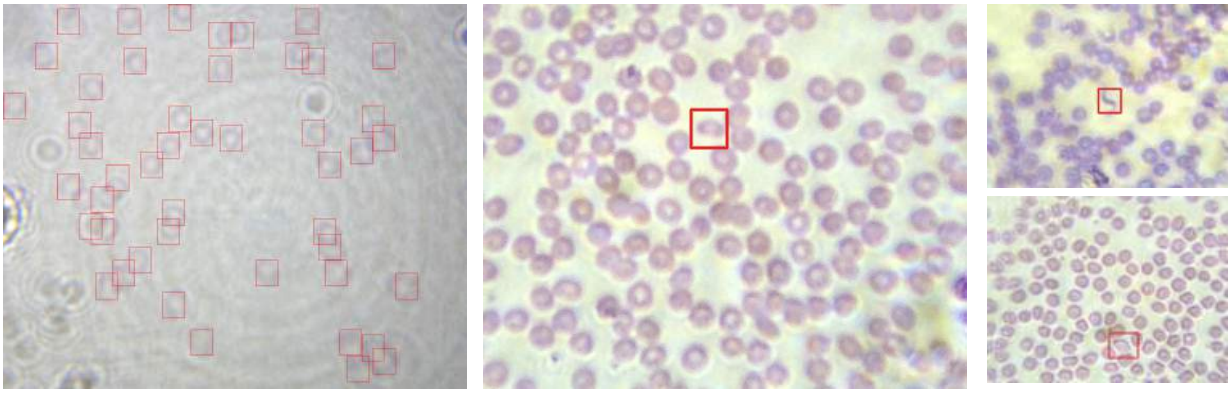
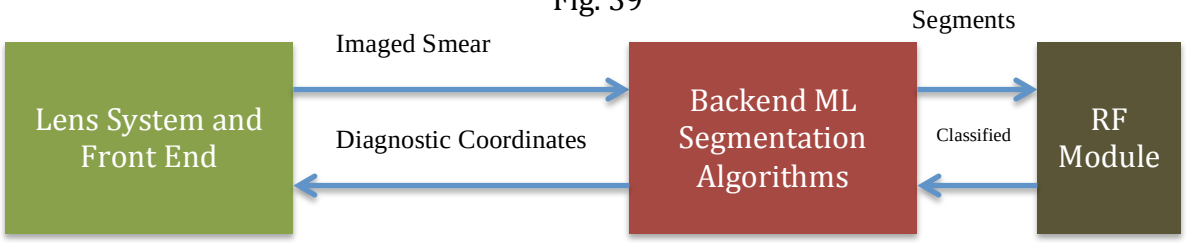


Figure 39 below highlight the backend model and the lens system integrating to identify cell types in the imaged smear. Essentially, the system is held above the sample, the software is run, and the diagnostic responses will be extracted by means of the backend topographical algorithm. The images above are from the system. Figure 35 highlights RBC coordinates identified by the algorithm on the phone, 36 highlights one of the identified sickle cell[12], 37 (top) shows an identified Trypanosoma parasite, and 38 (bottom) another sickle cell sample identification.

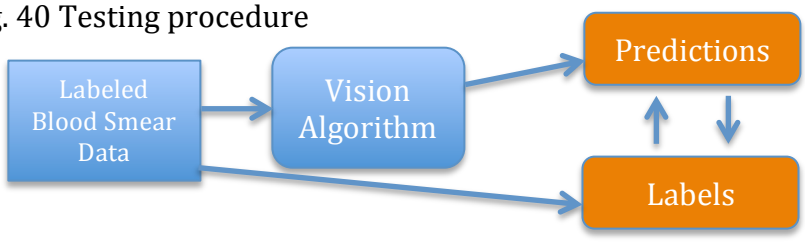
Fig. 39



III. Results and Data Analysis

The developed Topographical model for blood imaging and morphology was cross validated and tested for identification accuracy with reference to a human gold standard. For a given set of smear shots assigned to the computer vision system, the morphological labels for the cell were set aside, and the trained model would make a prediction. The outputted predictions were then verified in agreement to determine the accuracy of the model in terms in cell identification (see fig. 40 below). A test set of 1345 cell images randomly distributed between the 5 trained cell types was generated by imaging smear samples under the microscope. These test set samples were labeled in accordance to the procedure dictated in section 2.1.2.a and set aside for model cross validation. The RF model was then fed this unseen data to make predictions/classify the cell type, and the accuracy of the model with reference to the labeled gold standard was determined. For this test set of 1345 smear cells, the model classification accuracies are shown in Fig. 43 on page 19 bottom. A visualization of the model performance with respect to the number of training samples is shown in Fig. 36. The N_Samples are a randomized subset of the full training dataset (see Training table on page 10).

Fig. 40 Testing procedure



The outputted predictions by the vision model are cross validated against labels by human gold standard to determine performance. A test set of 1345 images were used.

Fig. 41 Sample to Prediction Accuracy Rate

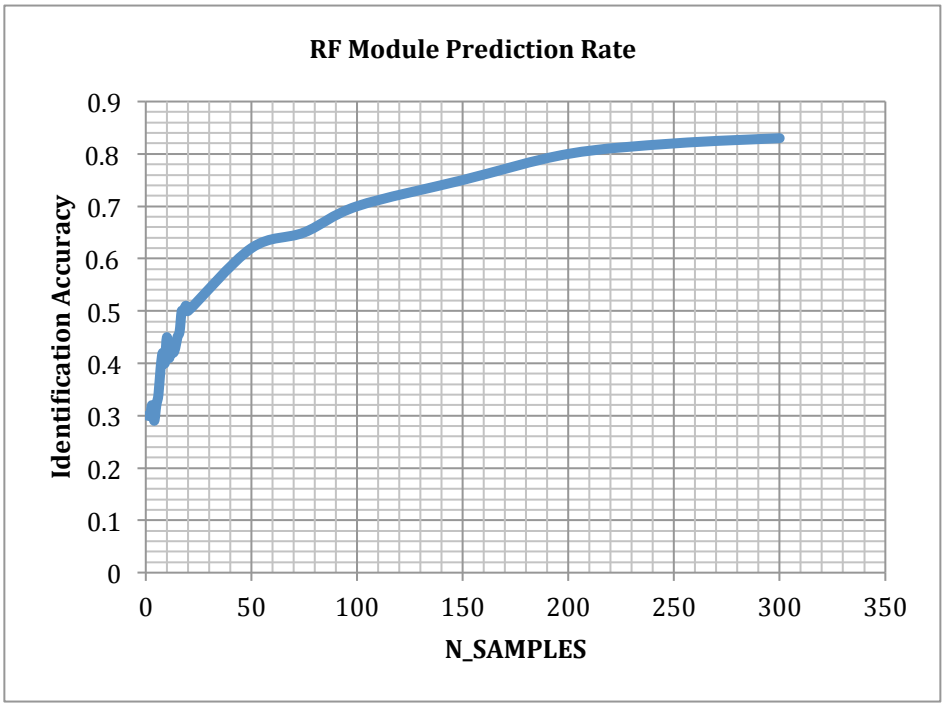


Fig 41 right. The number of training samples used to grow the Forest are shown relative to the Identification accuracy of the algorithm on a randomized subset of the data. As the number of samples increase, it is clear that there is an increase in the identification rate in the Random Forest. There is a clear tapering off (diminishing returns) in model accuracy towards increased training samples.

A test data smear classification is shown on the right. Red points indicate a classified Trypanosoma shot, and Green indicate erythrocyte. The axes are unit-less PCA derived components of the original 2500 feature space. The RF Module is provided a 10 dimensional version of this data (harder to visually depict in a graphical space) allowing for greater trend extrapolation for the Random Forest child trees. Below is a class-by-class accuracy assignment derived from the system testing and cross validation phase. **Human gold standards on direct analysis of smear are 0.9, and this data is relative to the original human morphology gold standard.**

Fig. 42 Smear shot in three dimensional PCA feature space

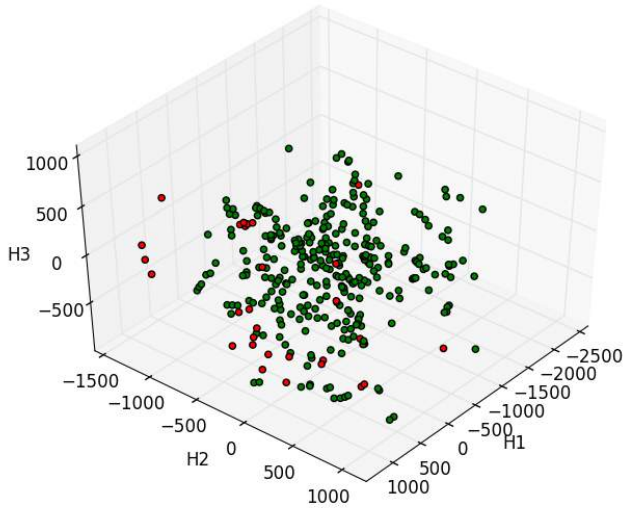


Fig. 43 Classification Accuracies

Sample Type	**Classification Accuracy**
Trypanosoma Cruzi	83%
Trypanosoma Brucei	80%
Drepanocytosis	76%
Erythrocyte	85%
P. Falciparum	73%

IV. Conclusions and Discussion

The algorithmic computer vision model and low-cost CMOS based lens system proposed in this paper provide a simple and trainable blood morphology test entirely portable on a smartphone for rural areas. The developed Topographical Euclidean distance transformation algorithm through multivariate local maxima peak analysis and PCA derived Random Forest classification successfully learn from training data at 73-85% cell accuracy identification rates. Furthermore, the engineered van Leeuwenhoek type lens-system magnifies smears by 360x with a 480uM field of view, interfacing with the morphological algorithm to recognize and classify parasite cells through a single ball-lens planted atop the camera in a frustum designed focal mount.

Due to paucity of clinical labs and skilled morphologists in underdeveloped regions, there is a high possibility of undetected diseases and delayed treatment. This device can be used in such rural areas to get portable blood test results similar to that of the skilled morphologist, thus reducing undiagnosed parasite cases and severity of condition at treatment time. The combination of the algorithmic study and optical lens engineering propose a highly viable solution to on-field blood diagnostics. By mimicking the procedure a trained morphologist takes in analyzing a blood sample by segmenting cells and individually conducting morphological analysis, the system successfully identifies parasites with an accuracy approaching human gold standard (0.9). Thus, the device and model would be invaluable to such underdeveloped regions where morphologists and large-scale setups are unavailable, by effectively running morphology in an automated, low-cost, and rapid manner.

Future work potentially involves assembling a much larger training set to push accuracy on the system to pass the human gold standard and utilize a Deep Learning model to improve classification accuracy. The novel proposed Random Forest based approach for cell image processing is resource and time efficient, however a potential Deep Learning approach may offer increases in accuracy. Due to the machine learning based nature of the classification algorithm, it is possible to upload training samples of dozens of other parasite smears, and immediately have the model ready to identify.

V. Acknowledgments

This study would not have been possible without the advice of Ms. Amita Kawale, Dr. Niaz Banaei of Stanford University, Dr. Ravinder Majeti of Stanford University, and Dr. Bryan Greenhouse of UCSF for their over-phone, email, and in-person mentorship regarding the microbiological portions of the research. By answering questions regarding parasite classification, cell morphology, current diagnostic gold standards they helped aid the process of engineering and developing the research behind this paper. Dr. Richard Socher of Stanford University also provided advice regarding computer vision and machine learning as a part of earlier research conducted under him in Natural Language Processing.

VI. Bibliography

1. Breiman, Leo (2001). "Random Forests". Machine Learning 45 (1): 5–32.
2. Smith ZJ, Chu K, Espenson AR, Rahimzadeh M, Gryshuk A, et al. (2011) Cell-Phone-Based Platform for Biomedical Device Development and Education Applications. PLoS ONE 6(3): e17150. doi:10.1371/journal.pone.0017150
3. The molecular phylogeny of trypanosomes: evidence for an early divergence of the Salivaria. Jochen Haag, Colm O'hUigin and Peter Overath, Molecular and Biochemical Parasitology, 1 March 1998, Volume 91, Issue 1, Pages 37–49, doi:10.1016/S0166-6851(97)00185-0
4. Sezgin and B. Sankur (2004). "Survey over image thresholding techniques and quantitative performance evaluation". Journal of Electronic Imaging 13 (1): 146–165. doi:10.1117/1.1631315
5. Garnham, P.C.C. (1966). Malaria Parasites And Other Haemosporidia. Oxford: Blackwell.
6. Malaria Fact sheet N°94". WHO. March 2014. Retrieved 28 August 2014.
7. Rosenblatt, F. (1958). "The Perceptron: A Probalistic Model For Information Storage And Organization In The Brain". Psychological Review 65 (6): 386–408. doi:10.1037/h0042519. PMID 13602029
8. Luc Vincent and Pierre Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. In IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, Num. 6 (1991), pages 583–598
9. Warhurst DC, Williams JE (1996). "Laboratory diagnosis of malaria". J Clin Pathol 49 (7): 533–38. doi:10.1136/jcp.49.7.533. PMC 500564. PMID 8813948
10. Gething PW, Elyazar IR, Moyes CL, Smith DL, Battle KE, Guerra CA, Patil AP, Tatem AJ, Howes RE, Myers MF, George DB, Horby P, Wertheim HF, Price RN, Müeller I, Baird JK, Hay SI (2012) A long neglected world malaria map: Plasmodium vivax endemicity in 2010. PLoS Negl Trop Dis 6(9):e1814
11. Madigan M; Martinko J (editors). (2005). Brock Biology of Microorganisms (11th ed.). Prentice Hall. ISBN 0-13-144329-1
12. Brown, Robert T., ed. (2006). Comprehensive handbook of childhood cancer and sickle cell disease: a biopsychosocial approach. Oxford University Press. ISBN 978-0-19-516985-0