Towards Automated Venipuncture

Rutgers University, Department of Electrical and Computer Engineering

James Chan and Philip Chan (Group 004)
Advisor: Dr. Laleh Najafizadeh

May 1, 2015

Abstract

Venipuncture, the drawing of blood, is susceptible to human error. Due to excess fatty tissue or loose skin, consistently finding veins can be difficult. Our project presents a step towards the automation of venipuncture in order to remove human error from the procedure.

The proposed device uses the principle of hemoglobin absorbing light in the near-infrared (NIR) spectrum allowing a camera to detect veins as dark spots on NIR-illuminated skin. Through image processing and a decision-making algorithm, the device will image the subject's arm, perform detection of veins, and make a decision on the best spot from which to draw blood. This system could potentially be installed at local clinics, providing for convenient and fast blood testing.

Contents

1 Introduction

1.1 Motivation

Venipuncture is used for countless tests, both diagnostic and routine. Despite being a simple procedure in theory, venipuncture can be problematic under certain conditions. Consistently finding veins can be difficult in patients with excess fatty tissue, loose skin, scarring, or decreased blood flow. This, in turn, can lead to multiple attempts to draw blood, wasting time and causing discomfort for the patient.

1.2 Proposal and Vision

We propose automating the process of venipuncture in order to remove human error from the procedure. In this project, we hope to provide proof of concept by implementing the camera module and computer vision necessary for automated venipuncture. The following steps are taken:

1. Skin is illuminated by NIR LEDs
2. A picture is taken
3. Vein detection is performed by the microprocessor
4. Decision on the best vein is made

In the future, the last step would be for venipuncture to be performed. This could either be done by displaying the output of the proposed system for a nurse as an aid for finding veins. More interestingly, venipuncture could be performed by a robot driven by the microprocessor. Eventually, we envision automated venipuncture becoming the norm, allowing for fast and reliable blood testing.

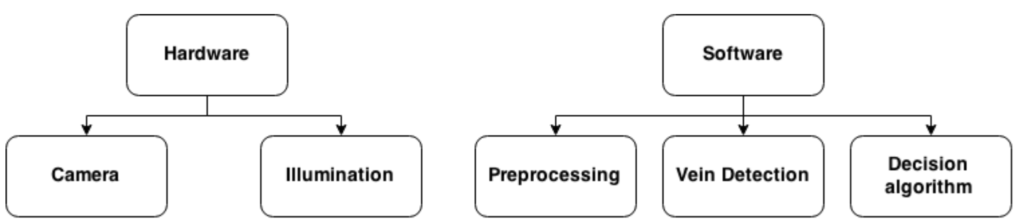

Figure 1: General block diagram of system

The project consists of two major portions, hardware and software as shown in figure 1. The hardware portion includes making decisions on which parts to use, making the actual device, and getting the illumination working to an acceptable level for the computervision to work. James Chan was in charge of the hardware aspects of the project. The software portion consisted of arriving at a spot on which to perform venipuncture based off of a picture of the arm. This necessarily included a preprocessing step for the image, a vein detection algorithm, and a decision making algorithm. Philip Chan was responsible for all software related aspects of the project.

1.3 Theoretical Considerations

Before taking a picture, the arm needs to be illuminated. The wavelength of light used used for this illumination plays an important role in producing clear images with detectable veins. Using visible light does not make sense, as it would not resolve veins more clearly or help to remove the error from venipuncture.

Looking at the absorption spectra of several chromophores in tissue, however, there is a clear choice of wavelength.

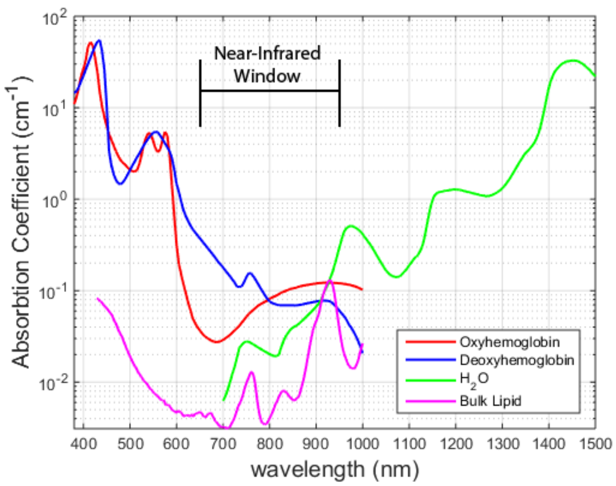

Figure 2: Absorption spectra of hemoglobin and other chromophores [1, 2, 3]

As shown in figure 2, wavelengths shorter than 650 nm tend to be absorbed by pigments such as hemoglobin and melanin. After about 950 nm, the absorption of light by water and lipids increases significantly [4]. Between these wavelengths, light has a large penetration depth into tissue, meaning subcutaneous structures such as veins can be resolved. This range is called the "near-infrared window in biological tissue" and is ideal for this application.

2 Method and Results

2.1 Hardware

The main purpose of the hardware used in this project is to provide near infrared illumination for the Raspberry Pi Camera. The hardware consists of the Raspberry Pi and the circuitry used for the illumination which is shown in figure 3.

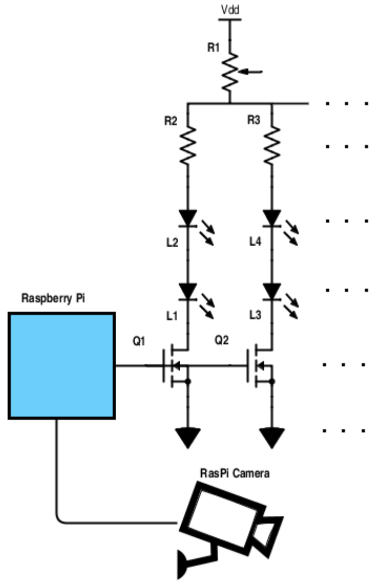

Figure 3: Hardware schematic

The circuit itself consists of a resistor in series with two near infrared LEDs and a MOSFET. For reasons previously discussed, we chose to use 850 nm NIR LEDs. The Raspberry Pi micro controller is attached to the gate of the MOSFET and controls whether the LEDs are turned on or off. A potentiometer was later added to vary the brightness of LEDs. We had a total of ten LEDs on the camera mount and thus the circuitry mentioned above was repeated five times for a total of ten LEDs.

The LED viewing angle was very acute and thus the area illuminated by a single LED was rather small. We chose to use a circular pattern for the LEDs, arranged around the camera in the middle. We made two camera/LED mounts with different diameters test this new illumination pattern. The larger one was four inches in diameter and the smaller one was two inches in diameter and both held ten, equally spaced LEDs. After testing, it was found that the larger one gave less even lighting distribution. The two inch diameter mount was chosen for the final project as we found the even lighting was better for illumination. Also, the smaller dimension allowed it to be more easily attached to the Raspberry Pi.

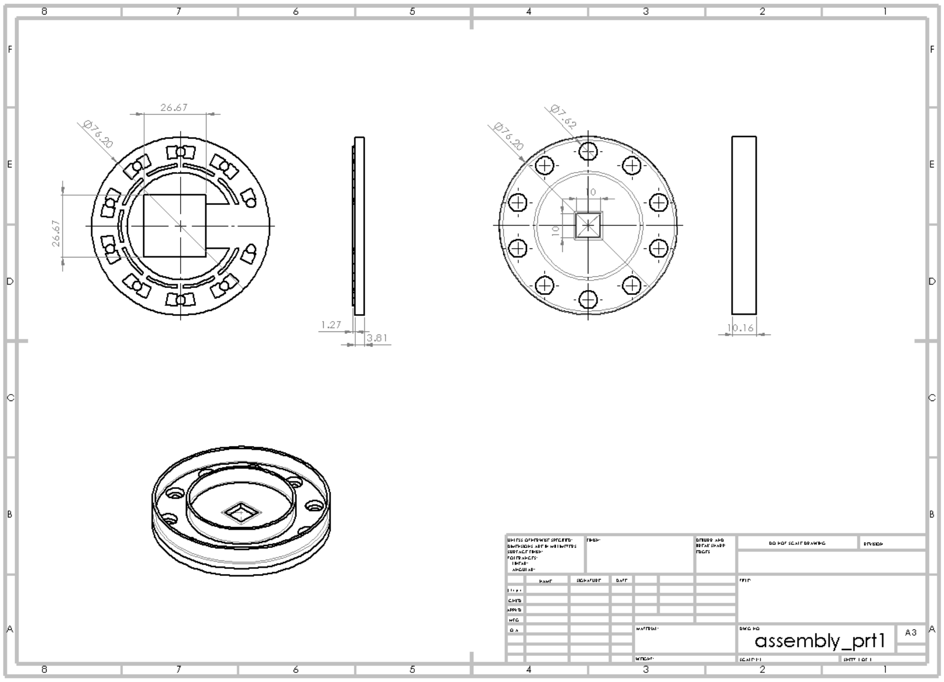

Figure 4: Schematic of the mount for LEDs and camera

For the final product, polarized sheets were placed over the LEDs and camera with a ninety degree separation in order to minimize the specular reflection. The actual camera mount was created in Solidworks as shown in 4 and printed using a 3D printer at Rutgers' Maker space. The final product is shown in figure 5.

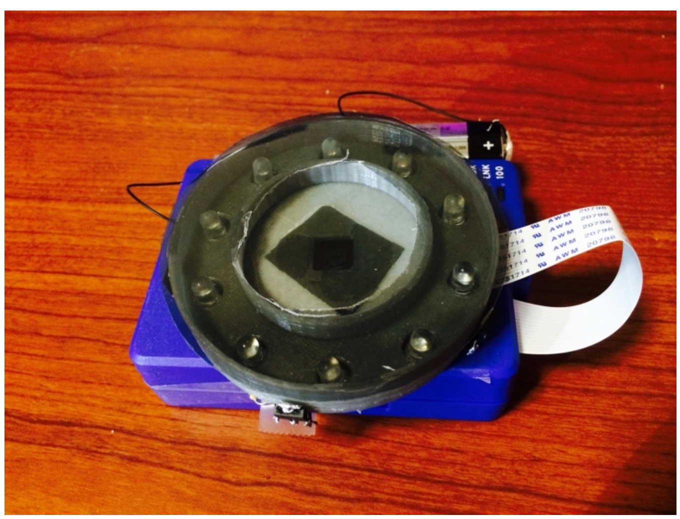

Figure 5: General block diagram of software

2.2 Software

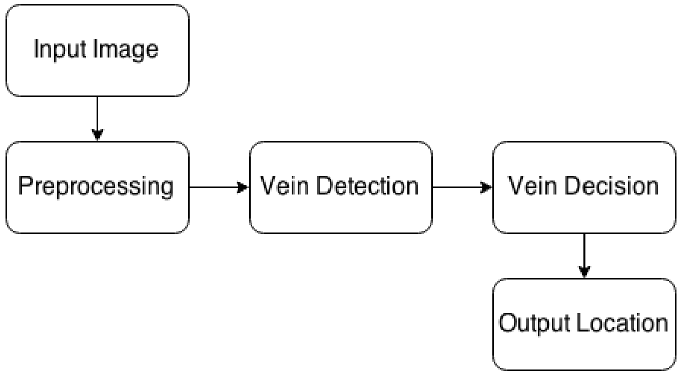

Figure 6: General block diagram of software

After receiving a picture of the arm from the camera, the software portion of the project consisted of three parts: preprocessing, vein detection, and decision making. The output of the program is a mapping of the veins in the arm and a location on which to perform venipuncture. Figure 7 shows an example of an unprocessed image from the Raspberry Pi.

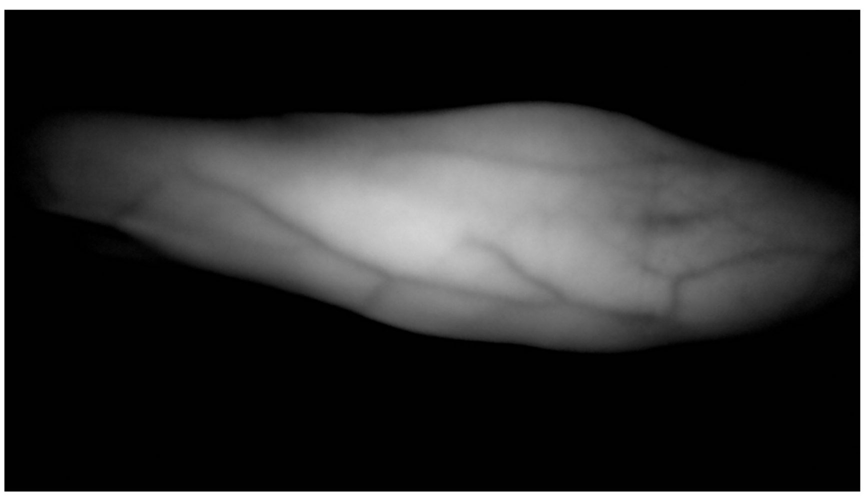

Figure 7: Example of an input image from the Raspberry Pi camera

2.2.1 Preprocessing

The preprocessing step prepares the image for vein detection. First, the input image is converted to grayscale and any background behind the arm is removed from the image by simple thresholding. Since the arm was illuminated with a single wavelength in NIR, color carries no useful information. Next, the contrast of the image is increased using the mapping $\mathrm{I} = 0.5(1 - \cos(\pi \mathrm{I}))$ where I is the image value; the darks are made darker and the lights are made lighter. This increases the difference in values between veins (dark sections of the image) and background. Finally, the image is blurred and down-sampled to remove some of the high-frequency noise (grain) and make the image processing faster.

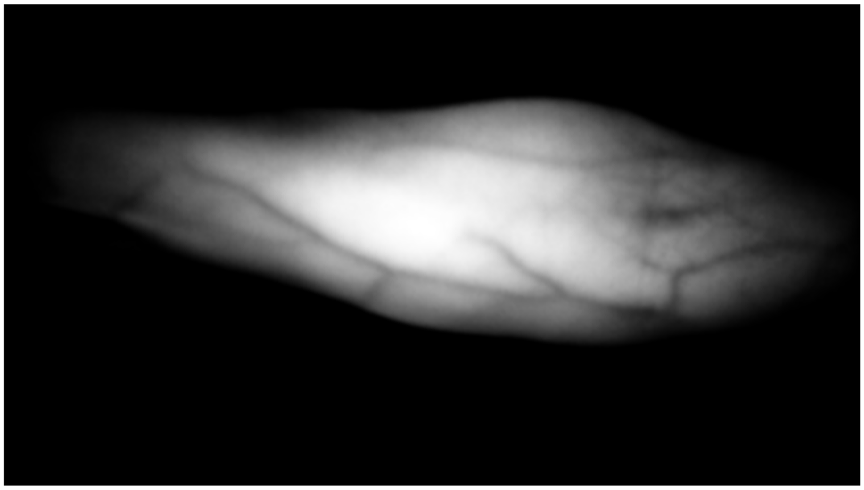

Figure 8: Image after preprocessing

2.2.2 Vein Detection

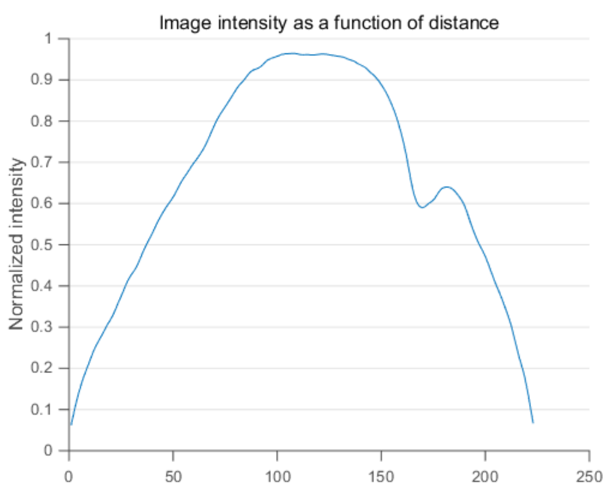

Figure 9: Vein intensity along a single column of the image

After preprocessing the image, the program attempts to locate the veins. Veins appear on the image as tracks of lower intensity, running down the arm. As shown in figure 9, veins are local minima in the columns of the image. The overall curve is due to the nonplanar surface of the arm causing uneven illumination.

The vein detection algorithm starts by choosing a set of seed points. These are chosen simply by scanning every n columns of the image for local minima and thresholding their depth as compared to the nearest local maximum. If a minimum is deeper than a threshold, the spot gets added to the list of seed points, otherwise it is ignored. As long as n is small compared to the length of a vein and the threshold value is low enough, this should seed a spot on every major vein in the image. We are looking to get a list containing several points in every major vein and few points laying outside of veins. Figure 10 shows an example of seed point selection on the sample image. As shown, most of the points fall on major veins with only a couple erroneously placed.

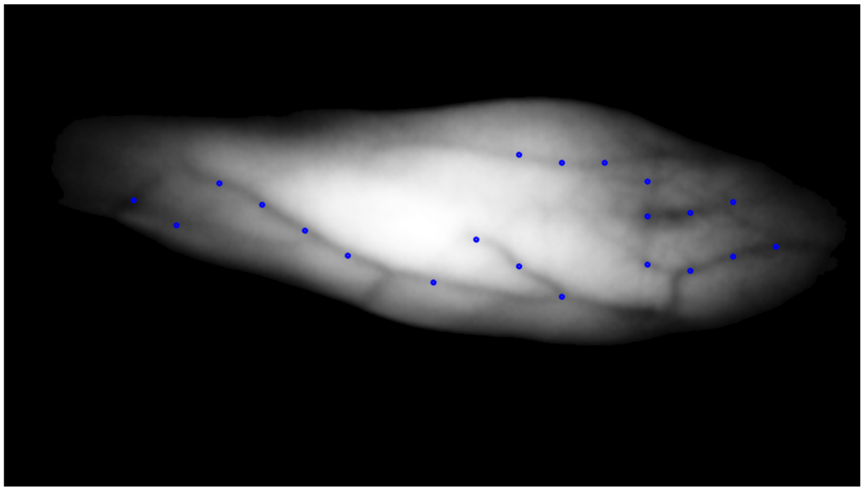

Figure 10: Image showing selected seed points

Once the algorithm has a list of seed points, it begins line-tracking. For each seed point, it attempts to follow the local minimum down the arm. Starting at a seed point, the algorithm moves its current position either diagonally up and to the right, directly to the right, or diagonally down and to the right, to whichever position has the lowest value. It then makes a check to see if a local minimum of sufficient depth to be a vein exists nearby in the same column. If so, the current position is moved to the local minimum and a vein-length counter is incremented. If not, the vein-length counter is incremented along with a no_check counter and movement is continued. If the algorithm goes through several iteration of movement without finding a local minimum (no_check $\geq$ threshold), it has probably reached the end of a vein or gotten off of track so the loop is exited. The vein-length counter is checked and, if it is sufficiently high, the path the algorithm traveled is added to a matrix and considered to be a vein.

As shown in figure 11, this algorithm is able to detect most large, lateral-running veins. Although there are several segments it was not able to detect, there are no false detections.

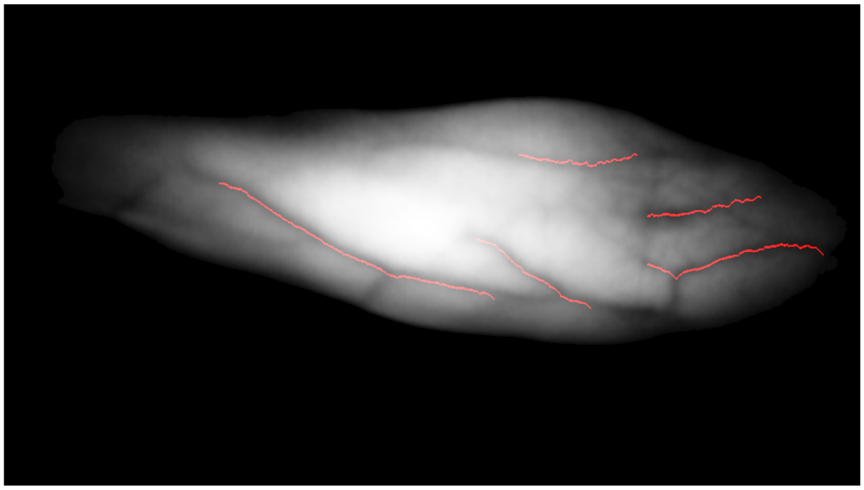

Figure 11: Image after vein detection

2.2.3 Decision

The veins have been detected and a decision on where to perform venipuncture needs to be made. This is done by keeping a weight value associated with points along the detected veins. The weight value is simply a linear combination of the depth of the local minimum (value of nearest maximum minus value of minimum) and the width of the vein (twice the distance from the nearest maximum to minimum). The coefficients were chosen experimentally until the expression yielded good results. As shown in figure 12, the traditional spot in the crease of the elbow was chosen.

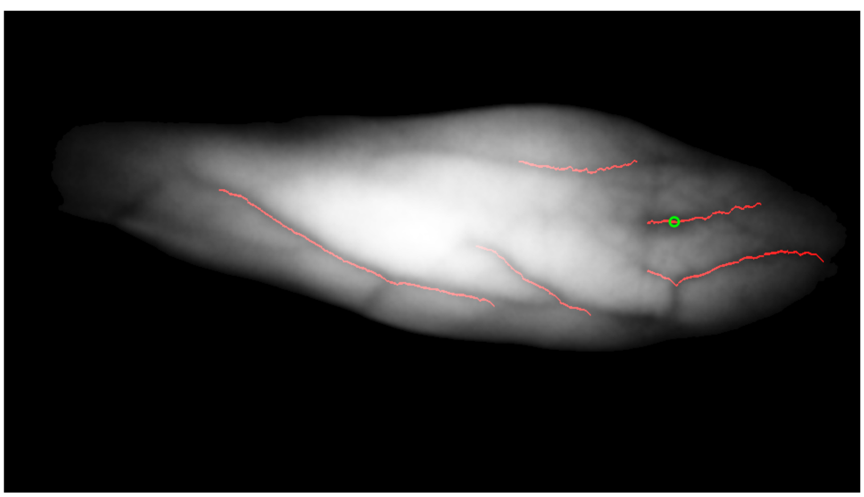

Figure 12: Image after decision is made on a location

2.3 Difficulties

2.3.1 Problems Encountered in Hardware

There were quite a few difficulties we encountered in the hardware section of the project, with the largest having to do with the quality of the lighting. At first, we believed that achieving the brightest lighting from the near infrared LEDs would produce the best image. This thinking led us to flash the LEDs very bright for a brief period of time. However, the bright LEDs caused a great amount of specular reflection off the arm, causing the image captured by the camera to be overexposed or "whited" out as shown in 13. The overexposed image made vein detection very difficult and nearly impossible.

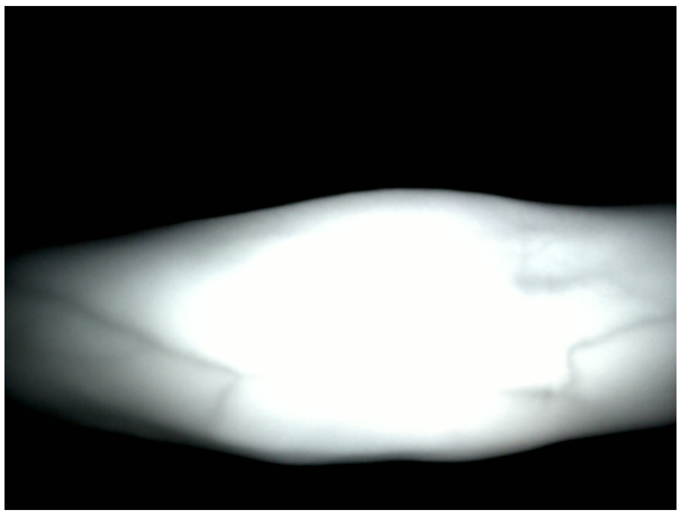

Figure 13: Example of overexposed image

The first solution that was proposed to minimize overexposure was to put some material over the LEDs to diffuse the light. The actual material used was scotch tape. The result was a dimmer picture of the arm, however the it was still rather overexposed.

Another solution was to vary the LED current with a potentiometer. The potentiometer would act as a variable resistor allowing us to vary the brightness of the LEDs to our liking. We found this change to be very useful as this allows for better lighting in different environments, although no ambient light is ideal. This also gave us great convenience as we no longer needed to physically change the resistors to vary the brightness. However, the original resistor was kept in place to prevent accidental shorting of the circuit which would damage the near infrared LEDs.

The second change that was made to remove overexposure was the installation of cross-polarized sheets over the camera and LEDs. The polarized sheets over the camera and LEDs were separated by a ninety degree angle for the purpose of minimizing the specular reflection off the surface of the arm. This solution helped immensely to produce useable pictures.

2.3.2 Problems Encountered in Software

In software, several attempts were made at vein detection before the current line-tracking algorithm was settled on. The first attempt was done through simple thresholding of local minima. The results are shown in figure 14. The thought behind it was that, if the conditions and illuminations were carefully tuned, then simple thresholding would give good detection. As shown, the detection of veins was fairly good, but it yielded a lot of false detections as well. Since this will be a medical device, noise and false detection is unacceptable.

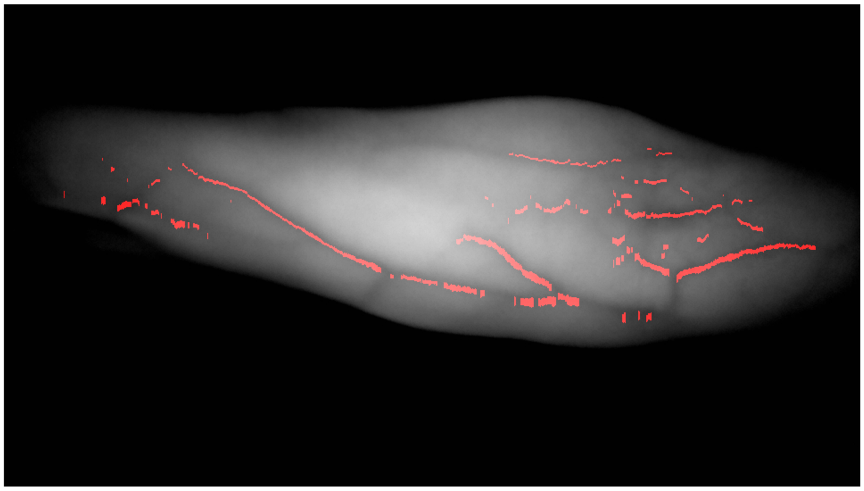

Figure 14: Simple thresholding of minima algorithm

With the current line-tracking algorithm, notice in 11 that only lateral-running veins are detected. Because the end-goal is venipuncture, this algorithm intentionally ignores any veins traveling transversely across the arm. Blood is generally drawn by moving the needle laterally into the arm, so it makes sense to only consider lateral-running veins to maximize the likelihood of successful venipuncture.

That being said, some effort was put into detecting transverse-running veins by repeating the algorithm rotated ninety degrees, i.e. seeding points that are local minima in rows of the image (rather than columns) and constraining motion to the up and down direction (rather than left/right). However, this runs into problems because of the large change in image values between the center and edge of the arm and tends to introduce error.

2.4 Results

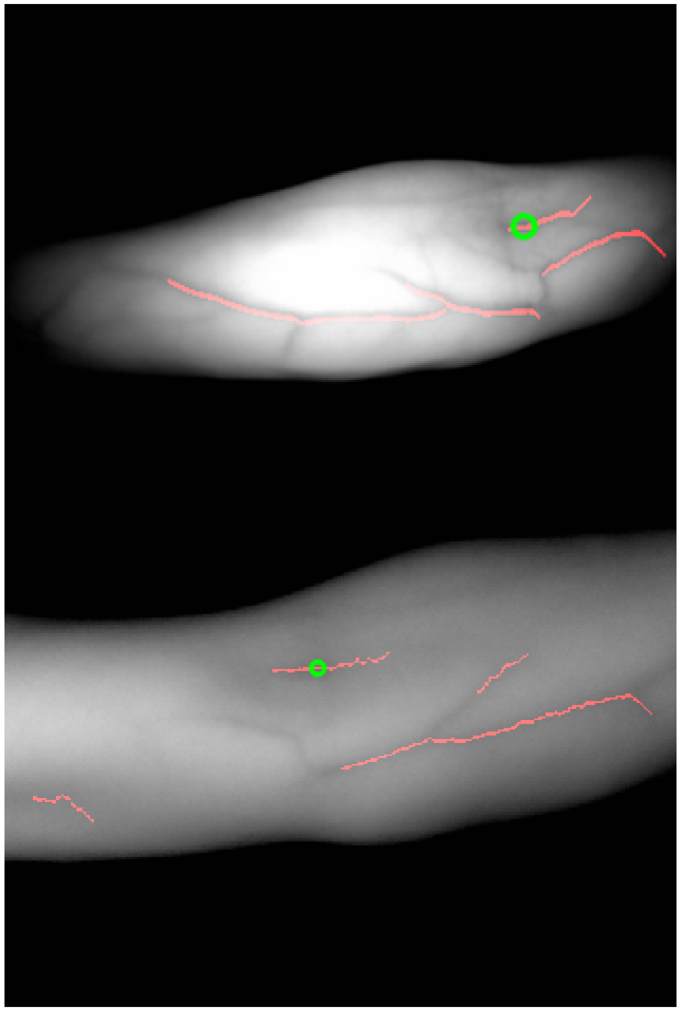

Figure 15: Example outputs

Figure 15 shows two more example outputs of the project. A command is sent to the Raspberry Pi, the LEDs are turned on, a picture is taken, and processed in GNU Octave. The processing detects veins and makes a decision on where to perform venipuncture based off of the intensity of the vein and the vein width. In all three of the results pictured (two in this section, one in the Software section), the tradition position in the crease of the elbow was chosen as the spot to prick.

2.5 Future Work

The final product of the project was proof of concept in the form of the camera module and computer vision necessary for an automated venipuncture. This camera module would be integrated with a robot capable of performing venipuncture.

In addition to integration with a robot, vein detection could be significantly increased by using multispectral imaging. Imaging in NIR gives information mainly about subcutaneous structures while imaging in visible light gives mainly reflection from the skin. Using multilspectral imaging would allow for the undesirable skin reflection to be subtracted out of the NIR image and thus increase the detectability of veins.

3 Cost and Sustainability

3.1 Costs

The total cost of this project was \$119.50. The main components were the Rapberry Pi microcontroller and the Raspberry Pi NoIR Camera which cost \$40 and \$33 respectively. The rest of the funds were speent between polarizing films, a microSD card, MOSFETS, and NIR LEDs.

As cheap as this project was, it merely represents proof of concept. The actual product would be integrated within a robot capable of performing venipuncture. The development of such a robot would introduce significant expenses. In addition, the software would need to be flushed out and made significantly more robust. While it works decently in a well-controlled environment, it would need to be able to track motion and have near-zero error.

Being a medical device, the product would need to be certified for actual use in the medical community. This certification is done by the FDA. First, the device needs to be classified into type I, II, or III depending on it's invasiveness and potential to do damage. Automated venipuncture would probably qualify as a new device, meaning a De Novo classification would need to be filed. As a new device, proof of reasonable safety and scientific evidence of effectiveness is required and the design is tested for worst-case failure, thermal failure, and package integrity [5]. Fees for qualification under form 510(k) or PMA range from \$5018 to \$250,895 depending on it's classification as either type II or type III respectively [6].

The actual realization of this proof of concept would incurr many costs. While the device itself may not be expensive, the cost of engineering it to be robust enough to pass medical regulations would be high. The computer vision would need to be extremely immune to error and the hardware would need to be packaged well. In addition, to fully automate venipuncture, an actual robot would need to be designed that could drive a syringe carriage to the correct position and perform venipuncture.

3.2 Social Impact

Automated venipuncture has the potential to greatly change the medical community. Currently, venipuncture is almost always performed by phlebotomists and nurses, making it subject to human error, especially in patients without visible veins such as infants. Nurses often struggle to find usable veins in these patients and may perform venipuncture in the wrong location leading to unnecessary bruising and pain to the patient. Automated venipuncture diminishes human error greatly and thus eliminates these problems of the past.

Furthermore, automating venipuncture will make the procedure more convenient for both the patient and the medical professional. In the future, patients may not need to go to the hospital to receive blood tests and could instead go to the local pharmacy, for example. This change is especially beneficial to rural communities with no hospitals nearby. Instead of long drives to the hospital, which might be located many miles away, the patient could visit the local pharmacy to get a blood test. Long lines in hospitals for routine blood tests would be relaxed and hospitals could allocate their staff to do other tasks.

4 Conclusion

Finding veins in certain individuals can be difficult. Therefore, we proposed the automation of venipuncture in order to remove this error. In this project, we demonstrate proof of concept for this by implementing the camera module and computer vision necessary for its realization. The device worked based on the concept of the near-infrared window in biological tissue, allowing the microprocessor to see veins more clearly than the naked-eye. Our final product was able to illuminate and image the arm, detect veins, and make a reasonable decision on where to perform venipuncture.

The trend in the health industry is towards automation. Already, many medical procedures that used to be performed by humans alone are now robot assisted. Paired with upcoming biosensing techniques such as lab-on-a-chip and BioFETs, automated venipuncture could be a powerful tool, allowing for accurate and convenient on-the-spot blood testing [7]. With this in mind, we believe venipuncture will follow this trend and will perhaps be implemented in the next few years.

5 Teammate Responsibilities

James Chan

- Decisions on components used in project
- In charge of Solidworks design for housing the camera and LEDs
- Soldering and physical creation of device
- Tuning of illumination through changing current, adding polarizer, etc
- Half of all presentations and reports

Philip Chan

- Aiding in Solidworks design and 3D printing
- Vein detection algorithm
- Vein decision algorithm
- Half of all presentations and reports

References

[1] D. Williams K. Palmer. Optical properties of water in the near infrared. *J. Optical Society of America*, 64:1107–1110, 1974.

[2] A. Pifferi A. Torricelli R. Cubeddu R. Veen, H. Sterenborg. Optical absorption of fat. Technical report, Oregon Medical Laser Center, 1999.

[3] S. Prahl. Optical absorption of hemoglobin. Technical report, Oregon Medical Laser Center, 1999.

[4] W. Want C. Tsai, J. Chen. Near-infrared absorption property of biological soft tissue constituents. *J. Med. Bio. Eng.*, 53:949–973, 2005.

[5] FDA Center for Devices and Radiological Health. *Design control guidance for medical device manufacturers*, 1997.

[6] FDA Center for Devices and Radiological Health. *Medical Device User Fee Small Business Qualification and Certification*, 2014.

[7] A. Poghossian M. Schoning. Recent advances in biologically senisitive field-effect transistors. *Analyst*, 127:1137–1151, 2002.