Technology Review and Implementation Plan

Core Body Temperature Estimation to Detect Ebola Virus Disease

CS 461, Fall 2017, Group 34

Brian Lee Huang

Abstract

The goal of this project is to be able to determine a person's core body temperature just by taking a thermal image of them, and analyzing the data take from that picture. This requires many parts, but the three I am tasked with is the Production code, Statistical Analysis, and the User Interface. To create these piece of the project I need to choose a pieces of technology to create them. For the production code I chose to use python, and analysis I chose to use R, a language specifically designed for statistical computing and analysis. For the user interface I am choosing to use command line as it is something easy to implement through print statements and command line arguments. However the choice for the user interface may change in the future depending on how far the project gets. The most important feature about all my choices is that each option is open source meaning there will be no licensing issues.

I. Introduction

The goal of this project is to create a program that uses a thermal camera to measure a person's skin temperature, then use that data to predict their core temperature. An elevated core temperature is an indicator that the person is symptomatic with the Ebola virus. The model and the analysis will be what predicts the core body temperature given a skin temperature.

For this project I am tasked with creating the Production code, User Interface and the Statistical Analysis. The production code and the analysis are mostly done towards the end of the project, while the user interface can be done at almost anytime during the project. The user interface should be very simple and not require much experience to use it. For most of the project a command line user interface will be sufficient.

II. Production Model

A. Overview

The production code is the second step, and what would be the final product of this project. As inputs it will take in a model, then take in a picture to then analyze the picture using the model provided. The output of this will just output a core body temperature.

B. Critera

The production code should try and be as accurate as possible, but should try to keep the false negative rate as low as possible. False negative should kept as low as possible as it can be extremely detrimental to misdiagnose someone that is symptomatic and putting them in the healthy group, as opposed to misdiagnosing someone that doesn't have it and putting them the unhealthy group.

We have found other projects that are similar to our project. They took a thermal image of a person's ear and head to derive a person's core body temperature. In this paper they achieved a positive predicted value of 40

C. Potential Choices

For this aspect of the project we have chosen three different possibilities to create a model. They are python, MATLAB and R.

D. First Choice

Python is a versatile language that we planned to use from the very beginning. It is easy to use, and has very many useful functions that makes data manipulation easier. It also has many plugins and libraries that can be used for model creation. We also plan to use python for our image analysis, so it will be easier to use the same language for both of these portions to make it easy to incorporate it.

E. Second Choice

MATLAB is a computing language created by MathWorks which is primarily used for numerical computing. This makes it an obvious potential choice to analyze the data from the image. MATLAB has many functions built into it so we mostly do not write any of the functions ourself for the analysis. The main difference between Python and R is that MATLAB requires a licence.

F. Third Choice

R is language used for statistical analysis and graphing. It can easily analyze the data taken from the image. R also includes many packages, so we can simply add in any feature that we need in the future. One small potential problem that we may run into with R is to have it run with our image processing program. We would need to set up a properly formatted text document as the output of our image processing and then run R with the document as the input.

G. Discussion

The first and most important distinction to make between the three choices is that Python and R are both open source, while MATLAB is not. This in itself makes MATLAB not a good option for us. Python is more of a general purpose language, while R has a specific use. For inputting data R is much better than Python because it is simply a few simple commands, while in python we will most likely have to write our own parser. Analyzing the data will most likely be easier in R as its intended use is to analyze data, while python is just general purpose.

H. Conclusion

Matlab requires a licence while the other two choices are free, so we most likely will not use MATLAB. We will most likely use python as our image handling program will also most likely be written in python. This allows us to just combine the program rather than handling the input and output of two different languages. However R will also be very useful for analyzing the data from the image, but incorporating the pre made model could be difficult.

III. Statistical Analysis

A. Overview

The statistical analysis is the intermediate step between the mathematical model and the data processing of the thermal images. We will use different statistical analysis to determine which is best for the creation of the mathematical model.

B. Critera

The analysis should show some kind of trend in the data. It should be easy to tell what different factors affect a person's core body temperature.

C. Potential Choices

For this aspect of our project we would use similar tools, if not the same tool for the mathematical model. Our choices for this piece are MATLAB, R, and Microsoft Excel.

D. First Choice

MATLAB can easily process huge amounts of data and create multiple different analysis of the data. It has many built in functions so the different analysis could simply be changed by changing a name, or a number of a function.

E. Second Choice

R is an obvious choice for as the whole point of R is statistical analysis. Just like MATLAB it can easily process huge amounts of data and create many different analysis. R also has many built in functions that do different statistical analysis to make the whole process easier. Data input and output would also be extremely simple with R as it simply uses a function call or two for data input and then another to process. Data output can also be setup to output to either a text file or an excel document.

F. Third Choice

Excel is another choice we had as it is a tool to create spreadsheets and analyze data. Data input is simple, and most of the analysis is simple. Output is also simple as it is all done through the Excel environment. No code has to be written and it is all done through the user interface of the program. The statistical analysis most likely would need to be done by hand, where we would need to set up equations using the cells as the data.

G. Discussion

In terms of power, Excel is lacking when compared to MATLAB and R. MATLAB and R are both stronger in than excel in the analysis portion as they both can import different libraries and packages if the built in tools are not enough. However both MATLAB and Excel require licences while R is open source. The versatility and the whole purpose of R make it the a clear choice for the statistical analysis. With easy input and output, being open source and its whole purpose being to analyze data.

H. Conclusion

R is the clear choice for this piece of the project as it is the most powerful and it is also free. Both MATLAB and Excel require licences that could become an issue in the future. Using an open source software makes it easier for us, and easier for anyone that intends to work on this in the future.

IV. User Interface

A. Overview

The user interface is what we will be using to input our data and display the output. In the end we need to create something simple that can be used by someone without engineering or computer experience. However for most of the project a extremely simple UI can be used. A nicer UI is only required if our project gets into its end stage, where it is almost ready to be deployed in the field. If we do not get to that stage, a graduate student may take over the project and for them I command line UI will be sufficient.

B. Critera

For our end product we need something simple and clear to use. The output of the program should be displayed with a simple yes or no. For the graduate student version, it should be simple to input data and have a clear output showing what it is doing. The usage of should be quick and clean so data input can be fast.

C. Potential Choices

For this piece of the project depending on where we are, we could use command line. However for the simpler version we chose to use either Java or Python. Command line is a good option because it can be written in almost any language and does not require any extra libraries or packages to install. Java and Python are also good options for a nicer looking UI because they have so many libraries and packages available to them.

D. First Choice

Command line is an easy choice for someone with a background in computer science. If a graduate student were to take over this project in the end a simple command line UI should be sufficient. A command line UI is easy to create, compatible with almost any language and easy to use, with a slight learning curve. Inputting data can be done with some command line arguments, while outputting can just be done with a couple of print statements.

E. Second Choice

Python is a good choice for UI design as it is easy and simple to use and there are many different libraries and toolkits available for python User Interface design. Python is also a good option for us in particular as we have planned to create the rest of the project in Python. This means that there should be no miscommunication between the languages when putting all the pieces together.

F. Third Choice

Java, just like Python has many libraries and toolkits built just for creating an user interface. One problem we may run into with using Java is the communication between the other parts of our project. We intend to create the other pieces using Python so creating a Java UI requires us have the two languages communicate which could cause issues.

G. Discussion

The command line interface will be easy to create as it can be created inside the main program.The other choices will have some learning required as the we would need research which toolkit we would want to use, and how to use the toolkit. However if we were to use one of the tool kits the user interface would be more robust and friendlier to the input and output of data.

H. Conclusion

Depending on where the project ends we can choose what kind of UI we want to make. For most of the project we will most likely be using a command line UI, where all the input and output is done from the program itself. In the end we will most likely create the UI using Python, as it will most likely be the easiest because we have already planned on using python.

References