*Nadai - Flood Detection and Gait Analysis using smartphone sensors*

*Change Name :*

*Poseidon - Water Depth Analysis using Gait and Smartphone Sensors*

**Abstract**

Floods are natural disasters common to places located near shores or which receive heavy rainfall. Floods are the most common natural disasters faced in the United States of America. They affect several aspects of society indiscriminately. During floods, transportation becomes risky, difficult, slow and cause a lot of inconveniences. They can lead to a major loss of life and property. Many a time due to incomplete and partial information people get stuck in traffics which are caused by waterlogging in the areas. These can be avoided or at the least minimized by careful planning and avoiding areas with higher amounts of water. This paper presents one aspect of planning a safe route for transportation during times of floods, which can also be used as a very good source of information for various rescue teams so that they consider the most affected areas initially. Using the sensors of mobile phones of people currently traveling in waterlogged areas, recording the depth of water in the area in which the user is traveling is done using computational statistical modeling. This data can be used to plan a safe route to travel. In order to accomplish this, data from Accelerometer, Gyroscope, Proximity Sensor, Gravitational and various other sensors<review.> is recorded. The acceleration from this data is in local coordinates of the mobile device.

**Intro - I**

1. **Motivation**

Being the home of over 1.2 Billion people, India is one of the fastest growing countries in the world. But the loopholes can be spotted in very basic aspects like city planning, which has to lead to a very common problem of drainage. Drainage is one of the ignition points of the of the floods. Transit from one place to another during the times of excess waterlogging and floods becomes excessively difficult. Even slight rains also sometimes create chaos in some areas. It is hard to predict in advance, which places will have more waterlogging than others. Monitoring flood event is necessary to analyze causes, methods for prevention and selecting the best path for transportation during emergencies and to help to respond efforts.

Thus motivation of this paper was to craft a simple, efficient and economical method to detect the depth of the water in flooded areas, which can be used as a guide to affected areas. There are only a scarce amount of readily available and useful techniques at the time of writing of this paper which could give such insights.

1. **Problem Statement**

*“Finding a reasonable method of detecting the depth/amount of water in the affected areas”.*

A method that’s inefficient of

Had there been a way to detect the same in earlier times, it’s solution could have been proved as an aid to the devastating floods like that of Kerala which happened a few weeks ago from the time of writing this paper.

1. **Suggested Solution**

A practical problem faced today by many people living in places with a high amount of precipitation or places which get flooded easily is the problem of waterlogging. Situations such as these often cause inconvenience and delays in transportation which is a major concern to the government, law enforcement, medical institutions, and business corporations. Delay in transportation can also sometimes cause major problems in situations of emergency.

Using sensors such as Accelerometer, Gyroscope, Proximity Sensor, Gravitational <review> and various other sensors present in the app AndroSensor, the height of the water in which the person with the smartphone is on foot is classified using a machine learning multi-class classifier. < Info. To be added about the model>. The information and the conclusion drawn from the findings and research on this paper can be used to practically deal with situations of flood and water logging in urbanized environments. One such solution can be to create a mobile phone application which will detect the height of the water user is present it. This can be analyzed by analysis techniques used by our study and can be uploaded to an online database. Information from this database can be accessed by other remote users through an interface and take appropriate actions.

This paper aims to analyze the waterlogged surrounding using general sensors present in which are very commonly held and present on hand during situations of waterlogging and flood.

**II Main Body**

1. **Data source and validity**

Data is the readings of various common sensors available in most of the smartphones. These sensors in include accelerometer, gyrometer, magnetometer<review> etc. It was collected using the mobile application *AndroSensor by Fiv Asim which is available on Google Play Store*.

Data collection is the most challenging part was done in various swimming pools with various depths. To collect the data, volunteers were made to walk in various depths. To maintain the simplicity of this work it was made sure that the android device was held in a common position for all the cases, i.e in hand slightly tilted near the belly( refer image <to be added>). The depths chosen for this experiment are

* 4.5 feet,
* 2.5 feet,
* 0.16 feet( 9 cm )
* 0 feet. ( Land )

The app[app reference] has been a very useful tool for completing this project. It also accommodates adjustable frequencies for recording and updating intervals. After experimentation, data was collected at a rate of 10 Hz, or 10 readings per sec. And updating interval set to very fast

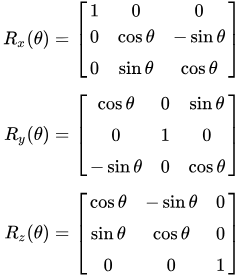
**Raw**

In praise

1. **Data preprocessing**
   1. **Rotational Transformation**

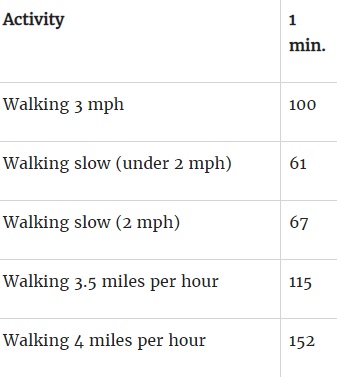
The attributes of acceleration have X, Y and Z components which were measured in respect with the X, Y and Z components of the device. This means the values of Acceleration in these 3 components would greatly be affected by the orientation in which the phone was being held. The analysis would benefit if these attributes were measured with respect to global axis X, Y, and Z. Values of acceleration which were measured in global coordinates wouldn’t be affected by the orientation of the device and would present a more general case. Hence, Rotational Transformation was conducted using the values of Azimuth, Roll, and Pitch present in Orientation X, Y, and Z.

Wikipedia: A basic rotation (also called elemental rotation) is a rotation about one of the axes of a 3 dimensional coordinate system. The following three basic rotation matrices rotate vectors by an angle θ about the X, Y, or Z-axis, in three dimensions, using the right-hand rule which systematizes their alternating signs. (The same matrices can also represent a clockwise rotation of the axes.Note that if instead of rotating vectors, it is the reference frame that is being rotated, the signs on the sin θ terms will be reversed. If reference frame A is rotated anti-clockwise about the origin through an angle θ to create reference frame B, then Rx (with the signs flipped) will transform a vector described in reference frame A coordinates to reference frame B coordinates.)



^Taken from Wikipedia

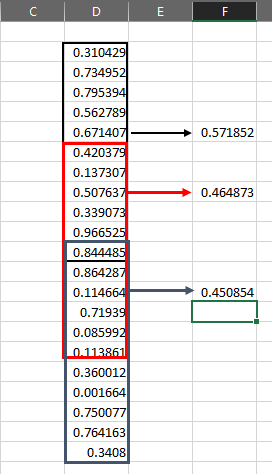
* 1. **Rolling Window Analysis**

Often for human activity recognition rolling/sliding window analysis is used. As it’s predefined in the name it calculates one value for a window of data, and that window keeps iterating over the whole dataset with a given amount of jump or stride. Various statistical methods are used for the calculation.

Since the problem here resembles a time series problem, a basic approach to solve them is using a rolling window analysis.

This image shows the average number of steps with the speed of walking. Which gives a simple relation of walking speed and number of steps.

Since while walking in different mediums aka different depths we tend to face different resistance on our legs which gives a change in the gait of the human.

But detecting this with point analysis could be very in convincing and accuracies were pretty imaginative. So the solution that seemed pretty obvious was to capture motion in a range of timeframe, which would give the model an overview of the gait. 

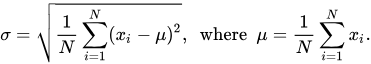
Thus, a window of 5 sec was chosen after hit and trial of many different window sizes. As mentioned earlier, the application uses a frequency of 10Hz for data collection which means 10 readings/per sec. Therefore,

So after calculating for 50 readings, the window moves ahead 50% ( chosen after experimentation and reference ), allowing data to be dependent on the previous window with not much of redundancy.

This image serves as an example for the above comments. For a series of data, a window of 10 is taken to calculate mean and then strode over with a jump of 5, giving 50% overlap.

Here are the following functions used for calculating various features in the dataset.

1. **Mean (µ) : ( Σ xi ) / n**
2. **Std Dev (**[**σ**](https://en.wiktionary.org/wiki/%CF%83)**) :**

****

**^copied from wikipedia**

1. **Median**
2. **Variance**
3. **FFT**
4. **Spectral Energy**
5. **Models**
   1. **Regression**
      1. **Motivation**
      2. **Logistic**
      3. **Polynomial**
   2. **Classifiers**
      1. **KNN**
      2. **Random Forest**

On the processed windowed data, a Random Forest Classifier model was trained. This model was built using the sklearn library in python.

* + 1. **Support Vector Machines**
  1. **Neural Networks**
     1. **One layer (both classifiers and regressions)**
     2. **CNN**
     3. **RNN**
     4. **Deep Feedforward networks**
  2. **Evaluation:**

For the evaluation of the model, 10 cross-validation approach was used in which the complete dataset comprising of 4 classes was split into 10 groups

**III References**

1. **Protter, Murray H.; Morrey, Jr., Charles B. (1970), College Calculus with Analytic Geometry (1970, p. 320, 2nd ed.), Reading: Addison-Wesley, LCCN 76087042**

**IV Citations**

**V Biblogrpahy**