Tongue-n-Cheek: Non-contact Tongue Gesture Recognition

Zheng Li
University of Maryland, Baltimore County
zhengli1@umbc.edu

Ryan Robucci
University of Maryland, Baltimore County
robucci@umbc.edu

Nilanjan Banerjee
University of Maryland, Baltimore County
nilanb@umbc.edu

Chintan Patel
University of Maryland, Baltimore County
cpatel2@umbc.edu

ABSTRACT

Tongue gestures are a key modality for augmentative and alternative communication in patients suffering from speech impairments and full-body paralysis. Systems for recognizing tongue gestures, however, are highly intrusive. They either rely on magnetic sensors built into dentures or artificial teeth deployed inside a patient's mouth or require contact with the skin using electromyography (EMG) sensors. Deploying sensors inside a patient's mouth can be uncomfortable for long-term use and contact-based sensors like EMG electrodes can cause skin abrasion. To address this problem, we present a novel contact-less sensor, called Tongue-n-Cheek, that captures tongue gestures using an array of micro-radars. The array of micro-radars act as proximity sensors and capture muscle movements when the patient performs the tongue gesture. Tongue-n-Cheek converts these movements into gestures using a novel signal processing algorithm. We demonstrate the efficacy of Tongue-n-Cheek and show that our system can reliably infer gestures with 95% accuracy and low latency.

Categories and Subject Descriptors

B.m [**Hardware**]: Miscellaneous

General Terms

Hardware Implementation, Sensor Design

Keywords

Tongue Gestures, Micro-radars, Paralysis Patients

1. INTRODUCTION

Paralysis caused by spinal cord injuries is on the rise in the United States and across the globe. In the United States alone, there are 250,000 to 500,000 new cases of spinal cord injuries registered every year [1]. Based on the severity of the injury, a patient might suffer from partial or full-body paralysis and limited mobility. For instance, a patient suffering from injury to the High-Cervical Nerves (C1-C4) have paralysis in their arms, hands, trunk, and legs [2]. Such patients can only use head, tongue, and eye movements as inputs for gesture recognition and environmental control. Amongst the three gesture inputs, tongue is a popular input organ for a large population of paralysis patients. This is due to two reasons. First, the tongue is a flexible organ and can be used to perform a large set of gestures. It is used for human speech and therefore is an intuitive gesture input organ. Secondly, the tongue is controlled by the cranial nerve [3] which is embedded in the skull and is not easily damaged during spinal cord injuries. So it is suitable for gesture recognition for a large population of paralysis patients.

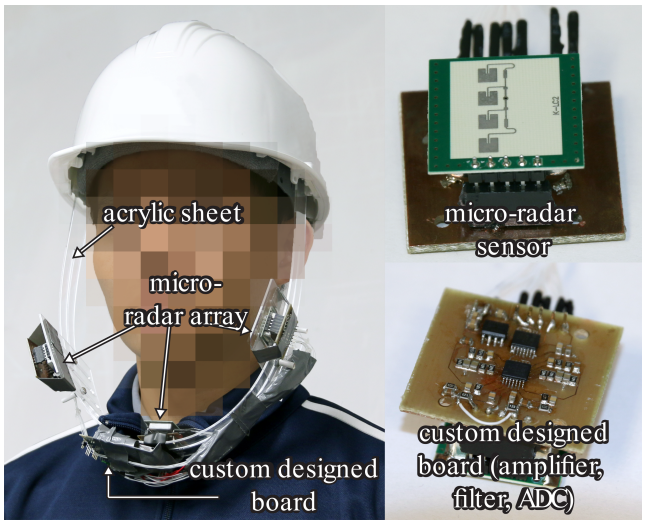


Figure 1: A prototype for the Tongue-n-Cheek system worn by a user. The sensor array comprises of three micro-radars bolted to thinly cut acrylic sheets attached to a helmet. We have fabricated a custom-designed PCB that houses a filtering circuit, an amplifier, and an analog-to-digital converter.

IPSN'15, April 13–17, 2015, Seattle, WA, USA.

The assistive devices built for gesture recognition and environmental control for severe paralysis patients using tongue gesture recognition, however, suffer from several drawbacks. For instance, state-of-the-art tongue gesture recognition systems use magnetic sensors that need to be placed into dentures or embedded into the tip of the tongue [4]. Such a sensor, although accurate, requires surgical deployment and is highly invasive. Similarly, researchers have proposed the use of infra-red sensors placed on the tongue which is equally invasive and intrusive [4]. A less intrusive sensor system for tongue gesture recognition uses the concept of surface electromyography (sEMG) that involves mounting the sensors on the skin surface close to the tongue [5, 6]. The sEMG sensors sensed and combined movement of suprahyoid muscles. Based on the relationship between tongue movement and suprahyoid muscles movement, the sensors can infer specific tongue gestures. While the system is non-invasive and less intrusive than sensors embedded inside the human mouth, EMG sensors are contact-based electrodes. These sensors use Ag/AgCl electrodes dipped into a polymer gel for better contact. The gel, however, dries out quickly and can cause skin abrasion. Since paralysis patients have low skin sensitivity, injuries caused by skin abrasion can be deleterious. Even more expensive dry EMG sensors can also cause skin abrasion. Completely non-intrusive systems such as Tongible [7] that uses a camera to determine tongue movements rely on the user opening their mouth when they performs a gesture—an action that may be difficult for severely paralyzed patients.

To address the above drawbacks in existing systems, in this paper we present Tongue-n-Cheek, a non-contact and non-intrusive tongue gesture recognition system. Our wearable prototype system mounted on a helmet is illustrated in Figure 1. Tongue-n-Cheek uses an array of micro-radars to detect movement of suprahyoid and cheek muscles. The system converts these movements into a set of reliable gestures using our signal processing algorithm. The design, implementation, and evaluation of Tongue-n-Cheek presents the following research contributions.

- **Non-contact tongue gesture recognition:** Tongue-n-Cheek uses an array of micro-radars to detect suprahyoid and cheek muscle movements correlated with tongue gestures. The system is completely non-invasive and minimally intrusive. The sensor is proximity-based and addresses skin irritation issues with contact-based tongue gesture systems that use EMG electrodes and invasive systems that use magnetic sensors embedded inside the mouth. The sensor array works in a collaborative way to cancel noise due to movements of the head and stray movements in the environment and can reliably detect movements caused by the tongue.

- **Signal processing on a micro-radar array:** Tongue-n-Cheek uses a signal processing algorithm that uses a combination of energy and velocity detection based on the phase difference between the I and Q channels for the received signal. These input signals are converted into gestures in real-time.

- **Prototype implementation and evaluation:** We have prototyped Tongue-n-Cheek, a fully functional embedded sensing-and-processing system, using three micro-radars that operate at a frequency of 24 GHz. We evaluate Tongue-n-Cheek using five subjects , and show that the microcontroller-based system can accurately detect gestures with a 95% accuracy in real-time.

2. RELATED WORK

Tongue-n-Cheek builds on previous work on tongue-gesture recognition systems, gesture recognition systems for paralysis patients, and micro-radar systems for healthcare. Here we compare and contrast Tongue-n-Cheek with the most relevant literature.

2.1 Tongue-Gesture Recognition System

Existing tongue-gesture recognition systems can be categorized into three broad groups. The most widely known tongue-gesture recognition systems use sensors embedded inside the mouth. For instance, the TongueDrive system [8] uses a tongue piercing to place a magnetic sensor inside the mouth, and an array of magnetic sensors outside the mouth. The array can infer the position of the tongue inside the mouth and hence can detect tongue gestures. The paper reported the system can achieve 90% accuracy in about 1s with a set of six gestures. Similarly, researchers have used infrared proximity sensors embedded inside the mouth to detect movements of the tongue with an average accuracy of 92.2% with four gestures [9]. Unfortunately, these systems are invasive. The second category of sensing systems use EMG electrodes to detect muscle movements when the tongue moves. The TongueSee [6] system detects movement in the suprahyoid muscles that is correlated to tongue gestures with 94.2% accuracy in 0.0625s with a set of six gestures. Other systems use the concept of surface electromyography (sEMG) that involves mounting the sensors on the skin surface close to the tongue [5, 6]. These sensors, however, are contact-based and can cause skin irritation that can cause injuries in paralysis patients. The third category of systems are completely non-intrusive and uses cameras to detect tongue [7]. The system requires the patient to open his mouth to perform the gesture, which can cause fatigue. Tongue-n-Cheek is a completely non-invasive and non-contact tongue gesture recognition system that addresses the shortcomings of the above systems by using micro-radars built into a wearable device. Moreover, our evaluation demonstrates that Tongue-n-Cheek has a significant improvement in terms of latency while reaching the state-of-art recognition accuracy.

2.2 Gesture Recognition for Paralysis Patients

There is a large corpus of research on building assistive-care devices for patients with limited mobility. These include the use of EEG sensors [10, 11] to detect brain waves for machine input, capacitive sensors [12, 13] to detect hand gestures, and eye-tracking systems based on cameras [14]. These different sensing modalities, however, do not focus on tongue gesture measurements. The goal with Tongue-n-Cheek is to use an array of micro-radars as proximity sensors for detecting movement of both cheek and suprahyoid muscles.

2.3 Micro-radar sensors

In our implementation, we use a RFBeam K-LC2 [15] sensor, a dual channel 24 GHz micro-radar. For our application, the micro-radar sensor should have the following two char-

acteristics that the RFBeam sensor satisfies. First, it should have dual channel (I/Q channels) output. With dual channel output, the sensor can distinguish movement direction and hence recognize a larger set of gestures. The details of analyzing the output I/Q channels is discussed in § 4. Secondly, the sensor system should be portable. A major component contributing to the size of the sensor is the size of the antenna. The single radiating element in the antenna is proportional to the wavelength of the transmitted wave. Our sensor operates at 24 GHz, hence, the wavelength is approximately 12.5 mm, so the sensor size is 25 mm · 25 mm. While a higher frequency micro-radar will increase the energy consumption and hardware complexity, a lower frequency radar will lead to a larger sized sensor. For example, the commercially available HB100 sensor manufactured by AgilSense operates at 10 GHz and is 4 times larger than our sensor [16].

Micro-radars have been used for several healthcare applications [17]. These include fall detection [18], remote heart-rate detection [19], respiration pattern detection [20], cardiopulmonary activity assessment [21], and tumor tracking [22]. These applications demonstrate that micro-radars are safe as a non-contact based sensor. The innovative solution presented in this paper uses these sensors for tongue gesture recognition.

3. GESTURE SET AND BACKGROUND

Our current gesture set comprises of six tongue gestures and one neutral gesture:

1. **Front:** moving forward
2. **Back:** moving backward
3. **Right-hold:** moving to the right and touching the right cheek
4. **Right-release:** moving from the cheek back to center
5. **Left-release:** moving to the left and touching the left cheek
6. **Left-hold:** moving from the cheek back to the center
7. **Neutral:** not moving the tongue.

These gestures are intuitively related to steering a power wheel chair for paralysis patients. Left and right tongue movements can be used for turning the wheelchair towards the left and right directions while the hold and release gestures can encode "keep turning" and "stop turning" respectively. The front and back tongue gestures can be used for moving the wheelchair forward and in reverse while no gestures correspond to placing the wheelchair in neutral. The above set of gestures can also be used for computer mouse control. The left-hold and left-release as well as right-hold and right-release gestures can be used to control the left and right button of the mouse, while the front and back gestures can correspond to the scroll wheel of the mouse. In addition to these seven gesture, Tongue-n-Cheek also supports a *wakeup* gesture: **Puff** which corresponds to filling the mouth with air and releasing the air, characterized uniquely by a simultaneous expansion on both sizes of the face. The puff gesture is a common gesture used in assistive care sip-and-puff switches.

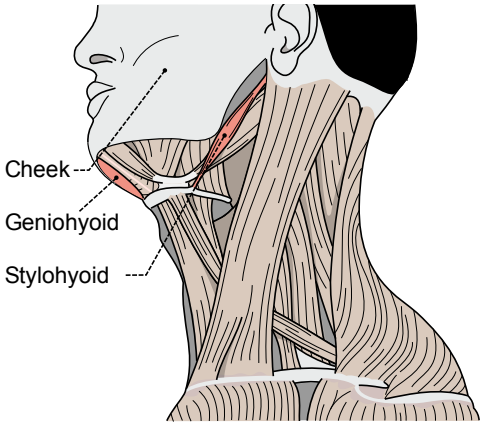


Figure 2: The muscles that control the tongue. In Tongue-n-Cheek we capture the movement of these muscles using an array of micro-radars. ©Olek Remesz

Table 1: The table shows the group of muscles that move when the six gestures are performed except Neutral

Front	Geniohyoid
Back	Stylohyoid
Right-hold	Cheek
Right-release	Cheek
Left-hold	Cheek
Left-release	Cheek

Figure 2 illustrates the muscles around the jaw and on the face. Table 1 shows the combination of muscles that move when the above gestures are performed. The fundamental contribution with Tongue-n-Cheek is inferring these gesture reliably using a touch-less proximity based sensor. Additionally, we demonstrate that Tongue-n-Cheek detects these gestures while canceling exogenous movements due to the head and other objects in the environment.

4. Tongue-n-Cheek HARDWARE SYSTEM

The physical sensors employed for gesture recognition are three micro-radars mounted on a wearable device. The radars sense movements using the principle of Doppler effect. The Doppler effect (or Doppler shift) is the difference between the observed frequency and the emitted frequency of a reflected wave when an object moves relative to the source generating the waves. Assuming the source and destination are co-located on the radar, the received frequency is higher than the emitted frequency when the object moves towards the radar. Inversely, the received frequency is lower than the emitted frequency when the object moves away from the source. In the presented application the movement measured is skin movement relative to the micro-radars. The skin movements detected are due to a combination of muscles when the tongue performs a gesture. The group of muscles that move when a tongue gesture is performed is described in Table 1. If f_r is the frequency of the received wave, f_t is the frequency of the transmitted wave, v is the velocity of the moving target object (collection of muscles in our case), and c is the speed of light, then the following

equation can be used to quantify the relationship between f_r and f_t:

$$f_r = f_t \cdot \left(\frac{c-v}{c+v}\right) \quad (1)$$

The shift in frequency is defined as $f_d = f_r - f_t$, which is approximately equal to $\frac{2 \cdot v}{c}$ when $v \ll c$, and can be used to determine the gestures. We empirically determined that velocity of movement of the muscles that correspond to our gestures lie between 0.02 m/s and 0.05 m/s. Since our radar works at a frequency of 24 GHz, the frequency shift we want to measure lies in the interval $[2.7, 8.0]Hz$. As our system detects tongue gestures in real-time, it needs to continuously determine the frequency shift (f_d) of the received wave from the transmitted wave. One straightforward solution to recover f_d is to use STFT(Short Time Fourier Transform) to get the spectrogram. However, the time resolution of the recovered f_d will be limited by the time-frequency uncertainty of STFT. Also, the computational complexity will increase the cost in terms of energy, memory, and latency. Another method is to demodulate the signal in analog and get f_d by calculating the instantaneous phase of the shifted signal. In the presented prototype, the received Doppler shifted wave is demodulated in the $I(t)$ (in-phase real component of the wave) and $Q(t)$ (quadrature or 90° shift of real components) channels. The Appendix shows the derivation of $I(t)$ and $Q(t)$ after applying low-pass filters.

An indicator of the phase difference between the transmitted and received signal can be found by taking the arctan of $\frac{I(t)}{Q(t)}$.

$$\Theta = \arctan\left(\frac{I(t)}{Q(t)}\right) = 2\pi f_d t + \phi - \frac{\pi}{2} = 4\pi\frac{vt}{\lambda} + \phi - \frac{\pi}{2} \quad (2)$$

Note arctan() is value in the interval $(\pi/2, -\pi/2)$, meaning an additive ambiguity of an integer multiple of π is added to the result. Section §5 contains a discussion on removing this ambiguity so that $\frac{d\Theta}{dt}$ provides the velocity of the movement that can be used as a distinguishing feature for inferring the gesture. This feature is used to design a robust signal processing algorithm described in §5.

To calculate the differential of Θ, however, the outputs ($I(t)$ and $Q(t)$) need to be amplified and filtered. This is because even with close distances between the radar and the moving muscle (close to 1 cm), the output of the I and Q channels still have a small energy of -60 dBm. To this end, custom printed circuit board was fabricated (Figure 1) with an analog bandpass filter (1 Hz to 60 Hz), an amplifier, and a 12-bit analog-to-digital converter (ADC) on the board. A ADC was added to the custom board instead of using the micro-controller ADC for two reasons. First, simultaneous capture of the data from many I and Q channels is not supported by the micro-controller. Secondly, the Tongue-n-Cheek micro-radar system has sensors placed at a distance from the micro-controller board and for prototyping purposes it was not desirable to transfer analog signals long wires since it requires careful shielding to minimize loss and noise.

Another issue with the system is the DC component of the signal. Even after filtering in both the demodulation and amplification stages, the DC component is likely to exist in the final analog output (input to the ADC) due to DC offsets. To address this problem, the offset was characterized experimentally and removed during the digital processing stage.

For our prototype implementation, RFbeam K-LC2 dual channel radars served as the sensors. Each sensor operated at 5 V and 32 mA in the experiments. As a control unit, an AVR Butterfly was used, which has a ATmega169P microcontroller with a 8 MHz clock and 1 KByte RAM.

To demonstrate that the prototype can capture subtle tongue gestures, Figure 3 (a) and (b) are presented. For this figure a subject performed the left-hold and left-release gestures. Figure 3 (a) shows the filtered and amplified I and Q channels waves when the left-hold gesture is being performed and differential of Θ, and Figure 3 (b) graphs the I and Q channels and the differential of Θ when the left-release gesture is performed. Figure 3(a) and (b) clearly shows that the I and Q lags and leads each other respectively for the two gestures. The figure also demonstrates that the differential of Θ captures this instantaneous phase difference.

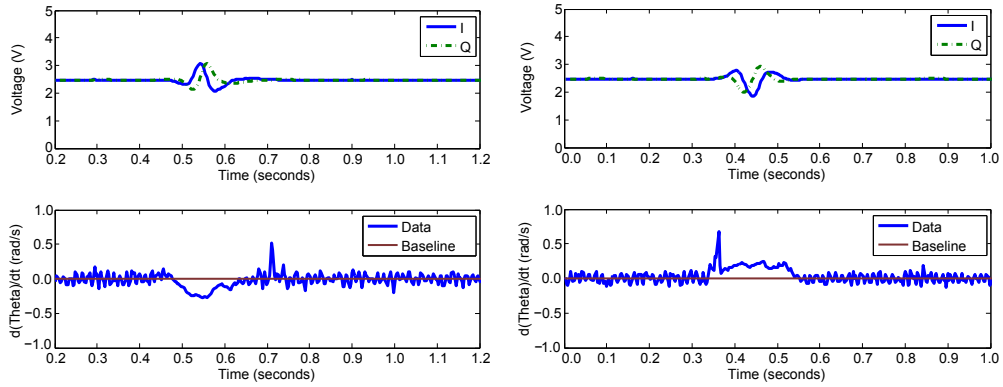
4.1 Micro-Radar Array

Micro-radars are directional and hence their cone of view is narrow (it is 30° in our prototype). Hence, capturing movement of a specific group of muscles would require directing the sensor towards that group of muscles. Our gesture set requires capturing movement of muscles on the left and right cheek, and the hyoid muscles below the jaw. To capture these movements simultaneously, we use an array of micro-radars. The *array* also helps address a critical challenge in our system–canceling noise due to movements of the head and stray movements in the environment. To illustrate the problem Figure 4 is presented, which compares the signature of the received waves when the user just performs head movements and when he performs head movements and tongue gestures when an array of two sensors are used. Sensor 1 in the figure is placed near the cheek and Sensor 2 is placed close to the forehead. The figure that taking the differential of Θ can help distinguish head movements from tongue gestures.

5. SIGNAL PROCESSING ALGORITHM

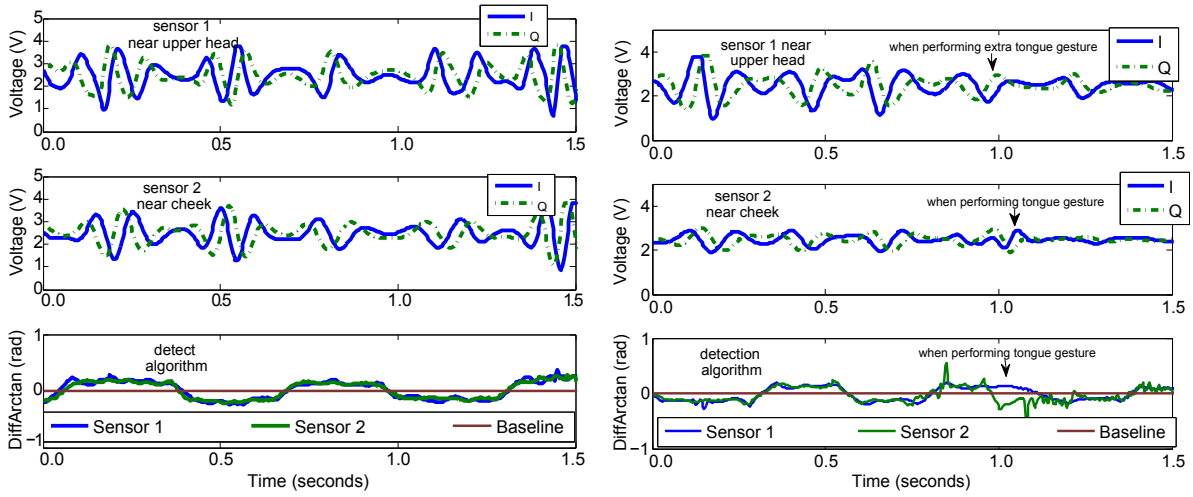
Figure 5 illustrates the overall system architecture of Tongue-n-Cheek. Data from the array of micro-radars demodulated in the I and Q channels is filtered using an analog bandpass filter and then amplified. The amplified signals are read by a micro-controller and converted into gestures. The key novelty of the signal processing algorithm co-designed with the physical system, described below, is in the optimizations to make it run in real-time minimal embedded platforms. Since an embedded platform is computationally weak, an important goal, therefore, is to minimize the computation-heavy operations such as multiplications and divisions.

There are two key steps in the Tongue-n-Cheek signal processing algorithm. First, the algorithm determines which micro-radar is detecting a gesture. It then segments the incoming I and Q data to determine the chunk of data where the gesture is being performed. Secondly, the algorithm extracts a single feature from the data chunk— the derivative of Θ defined in Equation 2. If the derivative is positive, then the muscle corresponding to the tongue gesture



(a) (b)

Figure 3: (a) The filtered and amplified output of the I channel and the Q channel, and differential of Θ when a user performs a left-hold gesture. Note that the I channel lags the Q channel, and the differential of Θ captures the movement. (b) The filtered and amplified output of the I and the Q channels, and the differential of Θ when the user performs a left-release gesture. Note that the I channel leads the Q channel, and the differential of Θ captures the movement.



(a) (b)

Figure 4: (a) I and Q channel outputs for two micro-radars, one placed near the forehead (sensor 1) and one placed near the cheek (sensor 2) when the head of the subject moves without any tongue gestures. The third subfigure shows the differentials of Θ for both sensors. (b) I and Q channel outputs for two micro-radars when the user moves his head and performs the left-hold gesture. The third subfigure that shows the differentials of Θ. The figure demonstrates that $\frac{d\Theta}{dt}$ can be used to distinguish head movements from the tongue gesture.

is likely moving towards the sensor (e.g., the **front**, **right-hold**, **left-hold** and **puff** gestures) or else the muscle is likely moving away from the sensor (e.g., the **back**, **right-release**, **left-release** gestures). It should be noted that in general multipath propagation should be considered as multiple reflection paths for the radar are likely. The summation of these paths result in a summation of several I and Q components dependent on the strength of the signal from each particular path. However, if it is assumed there is a primary movement surface generating a dominating return component then it can be assumed that $\frac{d\Theta}{dt}$ is due to that surface of movement. This assumption constrains the physical design of the system to minimize what the radar "sees" other than the desired skin area. In the algorithm, the direction of this primary movement (movement of the muscles towards or way from the micro-radar sensor) is found by determining the direction of change in $\Theta = arctan\left(\frac{I(t)}{Q(t)}\right)$ after applying a temporal phase unwrapping process. A description of the signal detection and segmentation, feature extraction scheme, and parameter training follows next.

5.1 Signal Detection and Segmentation

The signal processing algorithm in Tongue-n-Cheek receives the I and Q channel signals from the array of micro-radars. The first task that Tongue-n-Cheek performs is to analyze these signals and determine which radar is detecting a gesture. The radars are strategically placed such that a single radar points to a specific muscle group that moves due to a set of gestures. For instance, plausible sen-

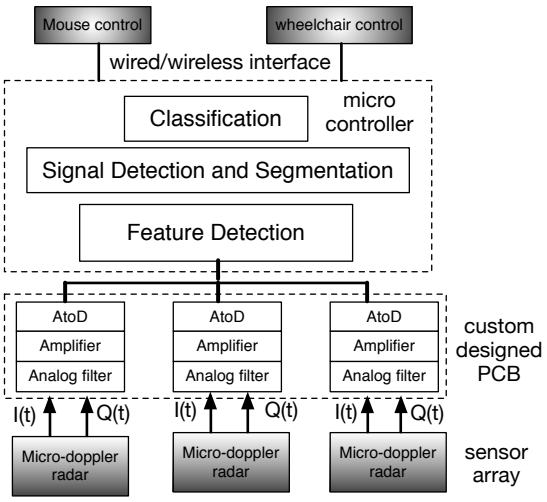


Figure 5: End-to-end system architecture for Tongue-n-Cheek. An array of micro-radars collect data on muscle movements due to tongue gestures and the received signal is demodulated in the I and Q channels. The signals from the channels are filtered and amplified and then read from a 12-bit analog-to-digital converter by the micro-controller. The micro-controller runs our gesture recognition algorithm that simultaneously analyzes the signals from the radar arrays and converts them into tongue gestures. The gestures can be used for wheelchair control, environmental control, or mouse control.

sor array configuration would have a radar pointing at the right cheek (radar 1), a radar pointing at the left cheek (radar 2), and a radar pointing at the lower jaw (radar 3). If a left-hold or left-release gesture is performed, then radar 2 will detect the gesture. For this detection, for every radar, Tongue-n-Cheek calculates the energy in the signal $E = \sum_{n=n_0}^{n_0+L}(I[n]^2 + Q[n]^2)$, where L is the length of a gesture defined in terms of the number of samples and is an input parameter for Tongue-n-Cheek. Note that the energy E in the signal is calculated on a moving window of length L. Once the energy E crosses a threshold δ_e (determined during a training phase), the segmentation module looks for a *maximum peak* in energy till the energy E falls below δ_e. If the center of this maximum peak occurs at time t_0, the time period for the gesture is taken from time $(t_0 - L, t_0]$. Figure 6 illustrates how the segmentation module works. This method of detecting when the gesture occurs has the following advantage. The smaller peaks, as shown in the figure, caused by stray movements in the vicinity of the primary muscle corresponding to the tongue gesture are filtered by the segmentation algorithm. Another issue is how to set the energy threshold δ_e. As the captured energy of tongue gestures vary based on radar position, helmet position, and users, a calibration phase is used to learn the threshold specific to the user and the helmet mounting. Detail of this training phase is described in §5.3.

5.2 Feature Extraction

Although Equation 2 demonstrates that the velocity of the muscle moving with respect to the radar can be determined

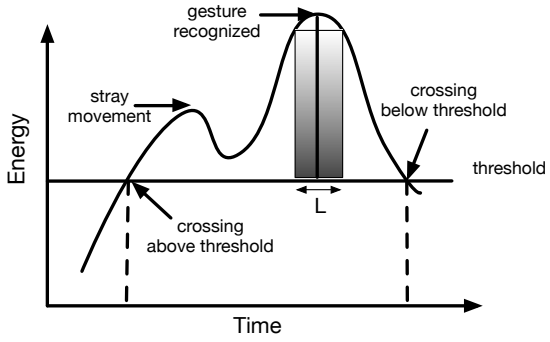


Figure 6: Signal segmentation. Tongue-n-Cheek calculates the sum of the energy in the I and Q channels using a sliding window of length L. The segmentation module looks for time instances when the energy crosses above a threshold. It then looks for energy peaks till the energy value falls below the threshold. The center of the maximum peak encountered is taken as the center of the gesture.

using a first-order derivative of $\Theta = \arctan\left(\frac{I(t)}{Q(t)}\right)$, in practice this velocity cannot be determined directly from arctan since it is not a one-to-one function. As arctan is bounded by $\pm\frac{\pi}{2}$ its temporal signal must undergo phase unwrapping. Therefore, $d(t)$ is calculated as follows.

$$d(t) = \text{phase\_unwrap}\left(\arctan\left(\frac{I(t)}{Q(t)}\right)\right) \tag{3}$$

The unwrapping is valid as long as we are sampling at least as fast as the Nyquist rate according to the maximum doppler shift frequency. The unwrap function is illustrated in Figure 7. However, calculating $d(t)$ requires evaluation of a trigonometric function which can be prohibitively slow to calculate on an embedded system. Note that this function has to continuously recalculated for realtime gesture recognition. Hence, it is important to optimize the calculation of this function to reduce the number of arithmetic operations.

There are a few methods available in the literature that allows approximation of the arctan function. For instance, arctan is commonly approximated by the following equation [23]. $\arctan(Q/I) \approx \frac{IQ}{I^2+0.28125Q^2}$. The scale multiplication by $0.28125 \times X$ can be implemented as $(X/4 + X/32)$ which can be implemented with two right shifts and an addition. In total 3 multiplications, 2 additions and a divisions are required to approximate the arctan reasonably over a range $\pm 45^\circ$ range. More expensive alternatives are Taylor approximations, while computationally cheaper alternatives include using interpolation on top of precomputed lookup tables.

In Tongue-n-Cheek the primary feature is the sign of the derivative of the phase unwrapped arctan. This sign is calculated using two multiplications and one comparison. The algorithm is based on the following intuition: the increase or decrease of the phase-unwrapped arctan can be predicted by the increase or decrease of the ratio of I and Q.

A computationally-inexpensive method for determining if $\frac{I}{Q}$ increases or decreases in a given timestep is presented here. The test $\frac{I_{n+1}}{Q_{n+1}} > \frac{I_n}{Q_n}$ can be rewritten by multiplying

each side of the inequality by the multiplicand $(Q_{n+1} \cdot Q)$. To account for the effect of the sign of this multiplicand on the inequality test we write the following, *ignoring the case where either Q or Q_n is zero*:

$$\left[\frac{I_{n+1}}{Q_{n+1}} > \frac{I_n}{Q_n}\right] \equiv \begin{cases} I_{n+1}Q_n > I_nQ_{n+1} & \text{if } sign(Q_n) = sign(Q_{n+1}) \\ I_{n+1}Q_n < I_nQ_{n+1} & \text{if } sign(Q_n) \neq sign(Q_{n+1}) \end{cases} \quad (4)$$

By assuming that

- the sampling rate at least as fast as the Nyquist rate so that one is able to perform phase unwrapping by ensuring that the phase does not change by more than π radians/sample.
- the amplitude is non-zero $(I^2 + Q^2 > 0)$ so that case when consecutive zero values in the denominator is avoided.
- $\frac{d\Theta}{dt} \neq 0$

, the two zero cases are sufficiently covered by

$$\left[\frac{I_{n+1}}{Q_{n+1}} > \frac{I_n}{Q_n}\right] \equiv \begin{cases} I_n < 0 & \text{if } Q_n = 0 \\ I_{n+1} > 0 & \text{if } Q_{n+1} = 0 \end{cases} \quad (5)$$

The system presented ensures that the above three conditions are satisfied during classification. The above equation outputs an array of +1 and -1 for a time window of the gesture determined by the segmentation module. To classify the entire movement as positive or negative (the sensor moving away and towards the radar), we employ a heuristic, which is a median filter to the array. The filter rejects sporadic high-frequency impulse noise [24]. From this we decide if the movement is positive or negative depending on the positive or negative result of the median. As there might be other muscle movement or head movement while performing the tongue gesture, we need to identify the sensor that captures the majority of the movement. Specifically, the left and right sensors are considered as higher priority sensors than the bottom sensor, as the suprahyoid muscle will move only slightly when a subject performs left/right hold/release gestures. Between the left and right sensor, the movement with larger normalized gesture energy is considered, assuming the intended gesture will generate a higher energy signal in the corresponding sensor. Then based on the combination of which radar detected the majority movement and the sign of the movement, the gestured can be uniquely determined. For instance, if the radar pointing at the left cheek detects the movement and the movement is positive, the gesture is left-hold.

5.3 Parameter estimation and training

The algorithm uses two parameters: L: the gesture length and δ_e: the energy threshold. In Tongue-n-Cheek these two parameters are determined during a training period. During the training period, a user performs each gesture k times. The system then performs peak detection on the energy calculated from the I and Q channels of the received data. δ_e is then calculated as the average of the minimum value of the negative peaks and the maximum value of the positive peaks. The optimal value of L is experimentally determined. In §6, the sensitivity of Tongue-n-Cheek with respect to L is evaluated.

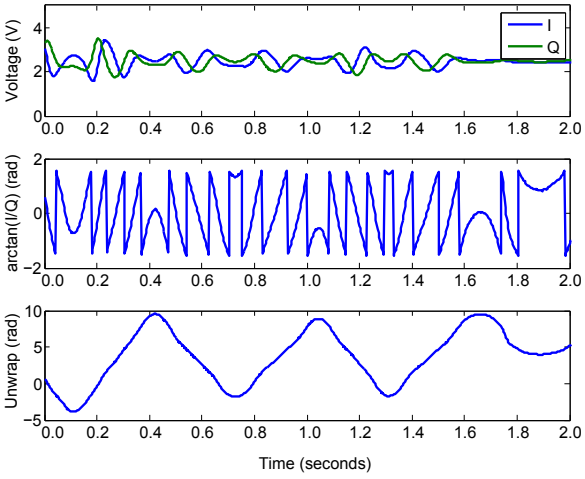


Figure 7: From top to bottom: $I(t)/Q(t)$, $\arctan(I(t)/Q(t))$, **and** $phase\_unwrap(\arctan(I(t)/Q(t)))$

6. SYSTEM EVALUATION

Our evaluation of Tongue-n-Cheek addresses the following key questions.

- How accurate is Tongue-n-Cheek in recognizing the seven gestures? What are the key causes of mis-classification of gestures?
- What are the trade-offs between system accuracy and amount of training required?
- How sensitive is the accuracy of Tongue-n-Cheek to the choice of L–the input gesture length?

Towards evaluating these key questions, presented also are results on the accuracy of the wake-up gesture, real-time performance of the signal processing algorithm, and micro-benchmarks on the micro-radar sensor.

6.1 Experimental Setup

Tongue-n-Cheek was evaluated on five test subjects. The subjects wear our wearable helmet with the sensors built into it (prototype illustrated in Figure 1). Since the helmet size did not immediately fit all subjects, it was adjusted for every subject. The subject was then asked to perform each gesture a small number of times (3 in all cases). The system calculates the energy threshold, δ_e for each micro-radar, using this training set. The subject was then asked to perform a set of gestures following instructions from a simple user interface that printed out the instructions. Each subject performed each gesture twelve times (eighty-four gestures in total). Though the experiments are stimulus-based, subjects may still have unintentional facial muscle movement and head movement during the experiment. These movement might cause false classifications. The subjects also performed the wakeup gesture multiple times. Results on system accuracy are presented next.

6.2 System Accuracy

Figure 8 presents a confusion matrix that illustrates the classification accuracy of Tongue-n-Cheek. The confusion matrix is built using all the gestures performed across all

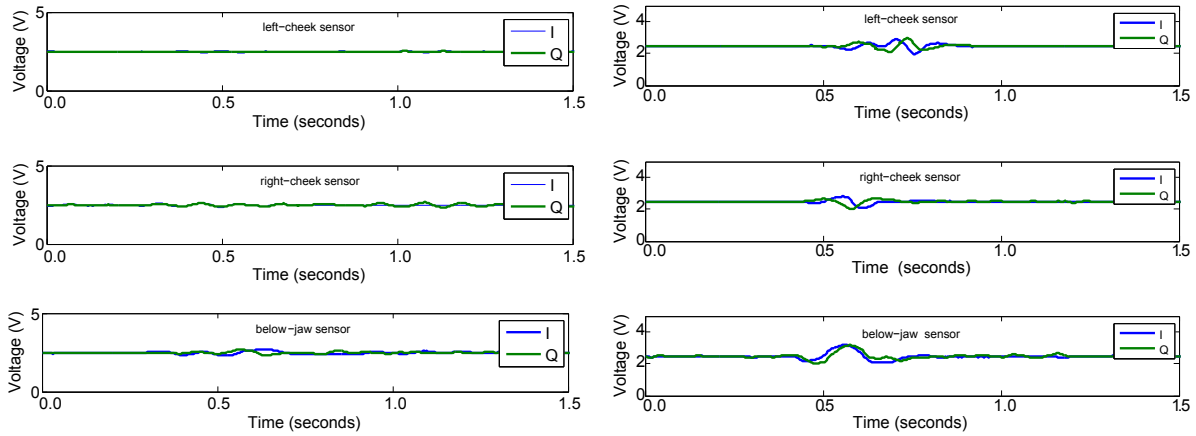


Figure 10: The figure explains the reason behind the misclassifications (a) Front gestures are misclassified as neutral since there are some instances when the user did not move his tongue forward enough for the energy threshold to be passed. The figure shows that the I and Q channel amplitudes do not change much when the gesture is performed. (b) Left-hold gestures are misclassified as right-release for a single user. This is because when the user performs a left-hold, the right cheek muscle is also stretched inwards causing the confusion. The figure shows that both the right-cheek and left-cheek sensors capture the two motions.

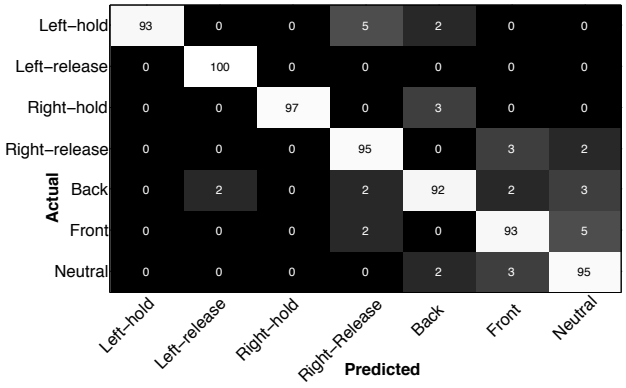


Figure 8: The figure shows a confusion matrix illustrating the classification accuracy of Tongue-n-Cheek. The confusion matrix is calculated by combining all the gestures across all the users (a total of 420 gestures). The average accuracy of Tongue-n-Cheek is 95%. Two of the highest misclassifications occur when left-hold gesture is misclassified as right-release, and the front gesture is misclassified as neutral.

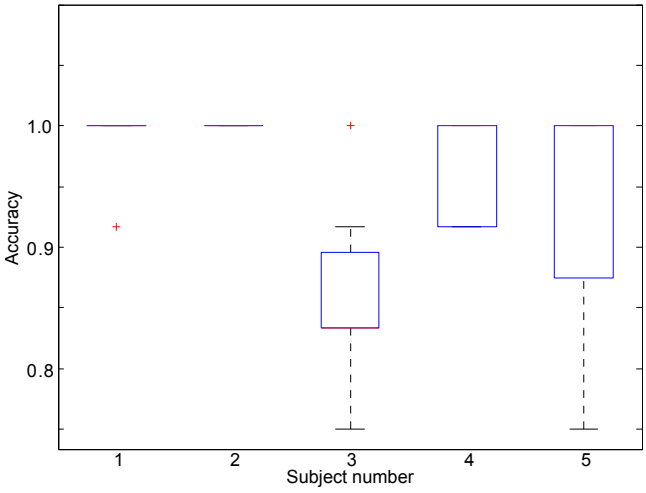


Figure 9: The figure shows accuracy of Tongue-n-Cheek per user. Tongue-n-Cheek can recognize gestures for User 1 and 2 with a 100% accuracy. User 3 has the lowest median accuracy of 83%. This is due to the misclassifications illustrated in Figure 10 (b).

the users, a total of 420 gestures. The average accuracy of Tongue-n-Cheek is 95%, which is higher than the accuracy of most contact-based tongue gesture recognition systems. For the result in Figure 8, the system was trained on a dataset where the users performed each gesture only three times. Hence, the training involved is minimal, making the system highly usable. Figure 8 shows that the left-release and right-hold gestures are recognized with an accuracy of 100% and 97% respectively. Most of the misclassifications occur in two cases: (1) the left-hold gesture is recognized as the right-release, and (2) the front gesture is misclassified as the neutral gesture. We further evaluate the underlying cause of these misclassifications in Figure 10 (a) and (b). Figure 10 (a) shows that when the front gestures are performed by one of the users, he does not pull his tongue forward enough to cross our energy threshold. In fact, the amplitude of the I and Q channel waves do not change at all, causing the misclassification. We believe that with proper training this problem can be addressed. Similarly, Figure 10 (b) demonstrates the case when the left-hold gestures are misclassified as right-release. This happens for a specific user because when he performs the left-hold gestures, the right cheek muscles are pulled in as well, causing the sensor on the right cheek to infer that the right-release gesture is being performed. The number of misclassifications, however,

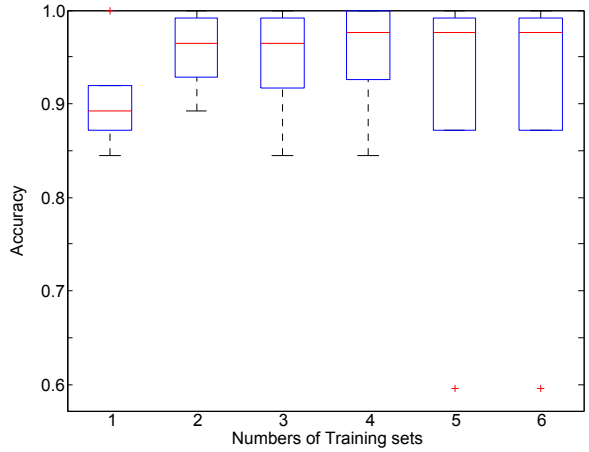


Figure 11: The figure shows how the accuracy of the system varies when the number of training sets is increased. One training set comprises of each gesture performed once. For this result, we calculate the average accuracy across all our subjects. The graph shows that the accuracy flattens after two training sets.

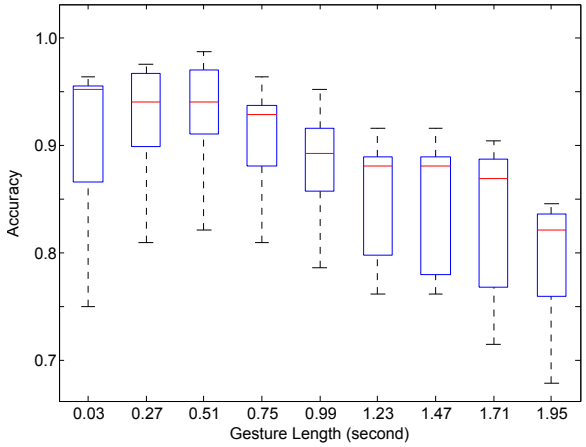


Figure 12: The figure shows how the accuracy of the system varies when the L, the gesture length, is varied. L is an input to our signal processing algorithm. The figure shows that as long as L is below the inter-gesture time, Tongue-n-Cheek's accuracy is above 80%.

are very few, only 5% of the total number of gestures on average, making Tongue-n-Cheek a very accurate tongue gesture recognition system. We further evaluate the accuracy of recognizing gestures per user (Figure 9). Tongue-n-Cheek can recognize the gestures from User 1 and 2 with a 100% accuracy. The median accuracy for User 3 is the lowest at 83%. This is because of the misclassifications discussed in Figure 10 (b).

6.3 System Accuracy Vs Training

An important trade-off in our system is how the accuracy of gesture recognition changes with increase in the training size. The training size influences the calculation of the energy threshold, δ_e. Figure 11 graphs the change in accuracy as the number of training sets used is increased from 1 to 6. Each training set comprises of one of each gesture performed once by a subject. We find that the median accuracy of recognizing gesture is more than 95% when only 2 training sets are used. This illustrates that the amount of training required by Tongue-n-Cheek is very low, make it highly usable.

6.4 Sensitivity to Input parameter L

We next evaluate how the accuracy of Tongue-n-Cheek changes with different L. Figure 12 graphs this trade-off. The figure shows that as long as the gesture length is lower than the time between performing gestures, our system accurately detects the gestures. When the gesture length L is more than the time between gestures (around 2 seconds in our experiments), our system captures more than one gesture in a one time window. Note that our signal processing algorithm uses the maximum peak to determine a *single* gesture in a period corresponding to one gesture length. When the system is deployed in practice, however, the frequency of performing gesture will be low so even a rough guess of L will produce high accuracy as long as L is below the inter-gesture time, making it a easy parameter to set.

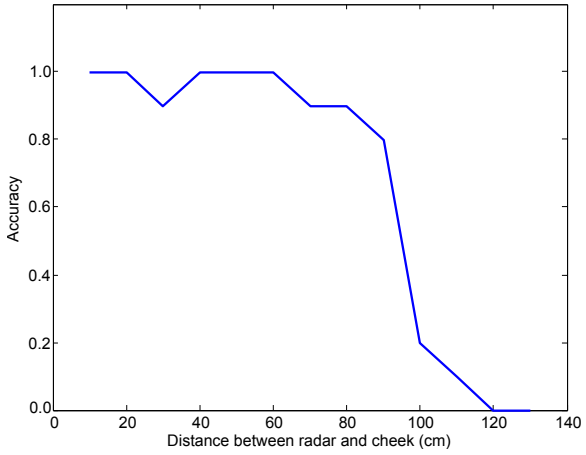


Figure 13: The figure shows the accuracy of recognizing the left-hold gesture as the distance between the radar and the cheek changes.

6.5 Micro-benchmarks

In this section, micro-benchmarks for the Tongue-n-Cheek system are quantified. Specifically, we quantify the accuracy of inferring the wake-up gesture, the latency associated with executing the signal processing algorithm on the microcontroller, and benchmarks on the range of the micro-radars. From our experiments we found that the accuracy of inferring the wake-up gestures is 98.2% and the system has few false positives in detecting the wake-up gesture. We also profiled the running time and the used memory of our optimized signal processing algorithm. For our prototype, we used a AVR Butterfly that operated at 8 MHz. Our algorithm utilized less than 800 Bytes of RAM and and takes less than 2.1 μs to analyze the I and Q data from the three micro-radar sensors and convert them into one of the seven gestures, demonstrating that the system can run in real-time efficiently. We have also evaluated the trade-off between the

range and accuracy of the micro-radar sensor. Figure 13 graphs the accuracy of recognizing the left-hold gesture as the distance between the cheek and the radar is varied. The figure shows that Tongue-n-Cheek can accurately detect the gestures even at a distance of 80 cm. Hence, the sensor array can be placed at a further distance, and it should still be able to infer the gestures accurately.

7. FUTURE WORK

There are several avenues of future work that we are pursuing as part of this project. We have detailed some of our future work below.

Explore other sensors

We are in the process of evaluating other micro-radars that operate at lower frequencies than 24 GHz. At lower frequencies, the skin penetration is deeper. However, the doppler shift will also be smaller. We also envision using ultra-sound sensor arrays for tongue gesture recognition in the future.

Larger sensor array

We are experimenting with a larger array of micro-radar sensors. With sensors strategically placed, the system can cancel stray and unexpected movements caused by wheelchair vibration or movements in the surroundings.

Larger gesture set

We are working on increasing the number of gestures in our gesture set. Moreover, we are working on designing a set of gestures customized to a user. Such personalized gestures are important, especially for paralysis patients where the acceptable movements might vary based on the specific injury type.

More user-friendly prototype

In terms of system usability, we are planning to build our prototype as an add-on to the wheelchair to get rid of the helmet. We are considering to evaluate our system on paralysis patients, as they might have some additional muscle movement while performing designed tongue gesture, such as twisting.

8. CONCLUSIONS

In this paper, we present Tongue-n-Cheek , a non-contact, minimally intrusive tongue gesture recognition system. This is unlike state-of-the-art tongue gesture recognition systems that either use contact-based EMG sensors or invasive magnetic sensors. Tongue-n-Cheek uses an array of micro-radars to detect muscle movements caused by tongue gestures. Signals received by the micro-radar array is converted into gesture using our signal processing algorithm that is optimized to run in realtime on a computationally weak microcontroller platform. We evaluate Tongue-n-Cheek on five subjects and show that it has an average accuracy of 95% in detecting gestures.

9. ACKNOWLEDGEMENTS

The authors would like to thank our shepherd Dr. Polly Huang and the anonymous reviewers for their insightful comments. This material is based upon work supported by the National Science Foundation under awards CNS-1305099 and IIS-1406626, CNS-1308723, CNS-1314024, and the Microsoft SEIF Awards. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or Microsoft.

Appendix

If the transmitted signal $T(t) = A_T \cos(2\pi f_t t)$, and the received signal $R(t) = A_R(t) \cos(2\pi (f_t + f_d) t + \phi)$, where ϕ is the phase related to the initial distances between objects in the scene, including the target surface, and the sensor. A_T and $A_R(t)$ are the amplitudes of the transmitted and received signals. Therefore, the I-channel wave equation is the following:

$$I'(t) = R(t)T(t) \tag{6}$$

$$= A_T A_R(t) \cos(2\pi f_t t) \cos(2\pi (f_t + f_d) t + \phi) \tag{7}$$

$$= \frac{A_T A_R(t)}{2} (\cos(2\pi f_d t + \phi) + \cos(2\pi (2f_t + f_d) t + \phi)) \tag{8}$$

The higher-frequency component at $2f_t + f_d$ Hz can be eliminated using an appropriate low-pass filter to recover $I(t)$:

$$I(t) = \frac{A_T A_R(t)}{2} \cos(2\pi f_d t + \phi) \tag{9}$$

Similarly, the received wave demodulated in the Q channel is the following.

$$Q'(t) = R(t)T(t) \tag{10}$$

$$= \frac{A_T A_R(t)}{2} (-\sin(2\pi f_d t + \phi) + \sin(2\pi (2f_t + f_d) t + \phi)) \tag{11}$$

After applying a low pass filter, $Q(t)$ is recovered:

$$Q(t) = -\frac{A_T A_R(t)}{2} \sin(2\pi f_d t + \phi) \tag{12}$$

10. REFERENCES

[1] *International Perspectives on Spinal Cord Injury.* WHO Press, World Health Organization, 20 Avenue Appia, 1211 Geneva 27, Switzerland, 2013.

[2] `http://www.spinalinjury101.org/details/levels-of-injury`.

[3] H. Kenneth Walker and W. Dallas Hall. *Clinical Methods: The History, Physical, and Laboratory Examinations. 3rd edition.* Butterworth-Heinemann, Boston, 1990.

[4] Xueliang Huo, Jia Wang, and Maysam Ghovanloo. A magneto-inductive sensor based wireless tongue-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(5):497–504, August 2008.

[5] M. Sasaki, K. Onishi, T. Arakawa, A. Nakayama, D. Stefanov, and M. Yamaguchi. Real-time estimation of tongue movement based on suprahyoid muscle activity. In *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pages 4605–4608. IEEE, July 2013.

[6] Qiao Zhang, Shyamnath Gollakota, Ben Taskar, and Rajesh P. N. Rao. Non-intrusive tongue machine interface. In *CHI '14 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2555–2558. ACM, April 2014.

[7] Li Liu, Shuo Niu, Jingjing Ren, and Jingyuan Zhang. Tongible: a non-contact tongue-based interaction technique. In *ASSETS '12 Proceedings of the 14th international ACM SIGACCESS conference on Computers and accessibility*, pages 233–234. ACM, October 2012.

[8] Xueliang Huo, Jia Wang, and Maysam Ghovanloo. Introduction and preliminary evaluation of the tongue drive system: wireless tongue-operated assistive technology for people with little or no upper-limb function. *Journal of rehabilitation research and development*, 45(6):921–930, 2007.

[9] T Scott Saponas, Daniel Kelly, Babak A Parviz, and Desney S Tan. Optically sensing tongue gestures for computer input. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 177–180. ACM, 2009.

[10] Jonathan R. Wolpaw and Dennis J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences of the United States of America*, 101(51):17849–17854, December 2004.

[11] Kevin R Wheeler and Charles C Jorgensen. Gestures as input: Neuroelectric joysticks and keyboards. *IEEE pervasive computing*, 2(2):56–61, 2003.

[12] A Nelson, J Schmandt, P Shyamkumar, W Wilkins, D Lachut, N Banerjee, S Rollins, J Parkerson, and V Varadan. Wearable multi-sensor gesture recognition for paralysis patients. In *Sensors, 2013 IEEE*, pages 1–4. IEEE, 2013.

[13] Alexander Nelson, Jackson Schmadt, William Wilkins, James P Parkerson, and Nilanjan Banerjee. System support for micro-harvester powered mobile sensing. In *Real-Time Systems Symposium (RTSS), 2013 IEEE 34th*, pages 258–267. IEEE, 2013.

[14] David Beymer and Myron Flickner. Eye gaze tracking using an active stereo head. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–451. IEEE, 2003.

[15] `http://www.rfbeam.ch/products/k-lc2-transceiver/`.

[16] `https://www.openimpulse.com/blog/products-page/product-category/hb100-microwave-sensor-module/`.

[17] Changzhi Li, V.M. Lubecke, O. Boric-Lubecke, and Jenshan Lin. A review on recent advances in doppler radar sensors for noncontact healthcare monitoring. *Microwave Theory and Techniques, IEEE Transactions on*, 61(5):2046–2060, May 2013.

[18] Liang Liu, Mihail Popescu, Marjorie Skubic, Marilyn Rantz, Tarik Yardibi, and Paul Cuddihy. Automatic fall detection based on doppler radar motion signature. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on*, pages 222–225. IEEE, 2011.

[19] Amy Droitcour, Victor Lubecke, Jenshan Lin, and Olga Boric-Lubecke. A microwave radio for doppler radar sensing of vital signs. In *Microwave Symposium Digest, 2001 IEEE MTT-S International*, volume 1, pages 175–178. IEEE, 2001.

[20] O Boric Lubecke, P-W Ong, and VM Lubecke. 10 ghz doppler radar sensing of respiration and heart movement. In *Bioengineering Conference, 2002. Proceedings of the IEEE 28th Annual Northeast*, pages 55–56. IEEE, 2002.

[21] Amy D Droitcour, Todd B Seto, Byung-Kwon Park, Shuhei Yamada, Alex Vergara, Charles El Hourani, Tommy Shing, Andrea Yuen, Victor M Lubecke, and O Boric-Lubecke. Non-contact respiratory rate measurement validation for hospitalized patients. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 4812–4815. IEEE, 2009.

[22] Changzhan Gu, Ruijiang Li, Steve B Jiang, and Changzhi Li. A multi-radar wireless system for respiratory gating and accurate tumor tracking in lung cancer radiotherapy. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 417–420. IEEE, 2011.

[23] Richard G Lyons. *Understanding digital signal processing*. Pearson Education, 2010.

[24] DRK Brownrigg. The weighted median filter. *Communications of the ACM*, 27(8):807–818, 1984.