Institute for Cognitive Systems
Department of Electrical and Computer Engineering
Technical University of Munich

Threshold-based Information Transfer System with IMU, EMG and Vibrotactile Motor

Ahmed Almijbari, Nathan van Beelen, Bertram Fuchs, and Runfeng Lyu

December 11, 2021

Abstract — EMGs, IMUs and Vibrotactile motors have been widely used in encoding and decoding number information. Hand writing and threshold-based encrypting techniques are two popular methods for ciphering number strings. Subjects only need to draw a certain sketch on a map as instruction for a machine to be recognized. Whereas, the transfer rate is still not ideal if numbers are sent continuously with short pause in between. In this article, we proposed a new method combining a threshold-based encoding strategy capable of encoding the numbers 0-9 to maximize the information transfer rate (ITR) without heavy pretraining for participants. An EMG signal is used as error-correction source to improve the accuracy on the encoding side. We also found that humans are more sensitive to visual and auditory vibrator signals compared to tactile sensory input on the skin. Hence, the receiver uses eyes and ears for decoding. Next to our methods as well as result analysis we will discuss the shortcomings and advantages of our proposed system.

1 Introduction

Vibrators have been widely used in perception for information transferring systems [3]. More recent research has also applied it with Brain-Machine Interfaces (BMI) to send and receive messages from different sensory patterns [1]. Vibrotactile motors are found to be a good guidance for blind people for accepting feedback during the interaction process with the world around them. In a similar way, it can also be applied to sending messages [9].This lead us to think about an easier way to transmit a message, namely by making use of an inertial measurement unit (IMU) device for encoding and vibrotactile motors for decoding.It does not require a complicated system setup process compared to EEG recordings.

The goal of this work is to create a system where information is transferred from one user to another by encoding the information using muscle movement and by decoding it using vibration. This work is part of a challenge. In this challenge, the system is limited to the following hardware: one IMU, one EMG, two Arduino boards, and four vibrotactile motors. The information to be transferred existed out of a list of 200 numbers (from zero to nine). The performance measurements of the system is the information transfer rate [4]. In other words, we tried to maximise the amount of information transferred over a certain amount of time. This means that a balance has to be found between the speed and accuracy of the system.

2 Methods

2.1 Human encoding

The encoding part of the system made use of an IMU and an EMG device. This allows the system to capture the user movement and muscle activity. The encoding of the system presented in this report was done using only the IMU. The EMG was used as an additional part of the system that allows for error correction.

The IMU measures from six different dimensions: three dimensions of acceleration and three dimensions of rotation. The system had to be easy to use and robust in encoding. Therefore, the number of dimensions actually used were limited where possible. This resulted in the following design for encoding.

The numbers one to eight were encoded using the 2D angle of acceleration. These angles were spaces as far away from each other as possible and kept in 2 dimensions to make movement as simple as possible (and thus classification as robust as possible). The number 9 was encoded by moving vertically and the number 0 was encoded by making a rotation. This keeps the encoding space between movement as big as possible while also keeping the movements themselves simple. The angle between all eight adjacent horizontal directions is 45°, the angle between all horizontal vectors and the vertical encoding of number 9 is 90°. Hence, a mean angle between two different encodings is 50°. A significant advantage to the following coordinate systems is the intuitiveness of this encoding scheme. The encoding scheme is illustrated in figure 1. In addition, figure 2 shows the encoding scheme used during experiments.

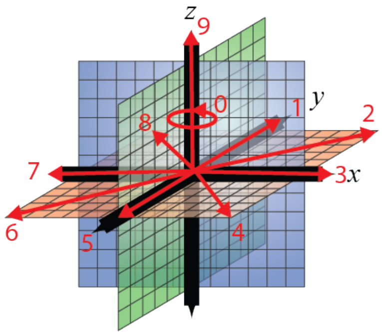

Figure 1 Encoding scheme. Numbers 1-8 are encoded in the xy-plane. Number 9 is encoded by a movement in z-axis direction. A rotation around the z-axis encodes number 0. After each number encoding, the IMU is moved back to the origin.

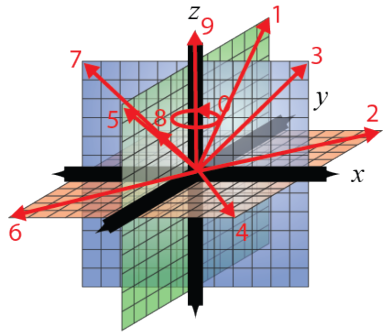

Figure 3 The Cartesian coordinate system constrained to positive z-coordinates is still intuitive. However, we expect hand movements containing both horizontal and vertical acceleration components to be shaky.

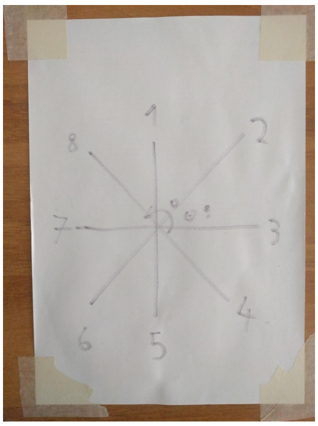

Figure 2 Paper guide used during experiments to make mapping directions to numbers easier. The IMU would be positioned in the center at the start of the experiment and moved in the direction of the desired number. Afterwards it would be moved back to the origin again.

2.2 Other considered encoding methods

2.2.1 3D half space movements

We decided against extending the linear movements from the 2D plane to 3D space as combinations of vertical and horizontal movements can not rely on a flat surface as support for stable hand movements. Our initial approach was to maximize the angles between each encoded number as much as possible. The mean angle between adjacent directions for linear movements for each corresponding encoded number is a good measurement for this. For eight horizontal linear movements and one vertical movement the angle between two adjacent hand movements can be computed with equation 1.

$$\phi = \arccos \frac{\vec{u} \circ \vec{v}}{|\vec{u}| \cdot |\vec{v}|} \tag{1}$$

In a Cartesian coordinate system which is constrained to a 3D half space as shown in 3 two adjacent movement directions correspond to the vectors in equation 2.

$$\vec{u} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \vec{v} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \tag{2}$$

In this configuration all vectors in the XY plane have a distance of 60° to the adjacent vector. The four vectors that include vertical and horizontal vectors are adjacent in this case to the vector parallel to the z-axis encoding number 9. These angles are all 45°. Hence, equation 3 gives the sum of the 9 angles as follows.

$$\phi_{total} = 4 \cdot 60^\circ + 4 \cdot 45^\circ + 45^\circ = 465^\circ \tag{3}$$

The average angle between adjacent directions would be 51.67°.

2.2.2 3D full space movements

If a whole Cartesian coordinate system is considered instead of one which is constrained to positive z-coordinates, the sum of the angles could be increased even further. Note however, that in both cases the encoding of the number 9 could be optimized even more and one would obtain nine vectors which are separated ideally, similar to a tetraeder, which optimizes the angle between four vectors in three dimensions. This optimization is called Thomson problem. Thomson tried to find the minimum electrostatic potential energy configuration of n electrons which all are located on the surface of a unit sphere under the condition that they repel each other with a force given by Coulomb's law [12]. Only for the cases $n = 1, 2, 3, 4, 5, 6, 12$ there

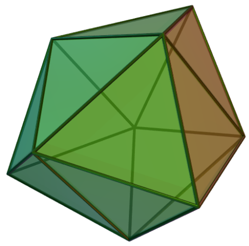

Figure 4 The mathematically optimal solution for maximizing the angle between all nine encoded linear directions is given by the Thomson problem. No exact solution exists for the case n=9. The numerical solution gives an angle of 69.19° for each pair of adjacent vectors. Due to the unintuitive resulting directions, we decided decided against this configuration. Instead, we decided for eight horizontal (numbers 1-8) and one vertical (number 9) encoding. The average angle between resulting vectors is 50°.

Linear movements	3D hand writing
0.47s	1.15s

Table 1 Average time needed per encoded number | In an experiment 100 numbers (numbers 1-9) were encoded and the time needed was measured. The time needed for encoding a linear movement is less than half the time needed for encoding hand written numbers.

exist exact solutions. For n=9, which is especially interesting for us, the angles between all nine directions are 69.19° and must be calculated numerically. This configuration is called triaugmented triangular prism. An illustration is shown in figure 4.

2.2.3 Writing numbers

As we have shown, intuitiveness is an important aspect of the encoding scheme. Hence, we thought about schemes which are even more intuitive than our proposed planar plus z-axis approach. The idea of encoding hand movements came up. Several aspects spoke against that approach. First, the duration of linear movements is much lower. For getting quantitative values we took the time needed to encode the numbers 1-9 both with linear movements and with hand writing movements in the air, writing the digits until hundred written numbers were reached. Table 1 shows the resulting times.

Furthermore, we assumed that linear movements all take the same time and have a similar acceleration profile which makes recognizing a reference origin point easier and faster compared to numbers which might differ in time needed for encoding. Encoding numbers with linear movements allows using a simple threshold method. For the hand movement encoding we expect that a classifier is necessary for decoding. The constrained computing power of the Arduino needed for this task is also expected to be high, which is a further disadvantage of this method.

Next to the intuitiveness of the in-air hand movements might be the accuracy when a good classifier is used. However, we expect that new reference data is needed for training the classifier for every new participant. In order to achieve a high ITR, we wanted to take a neuro-centered approach and hence tested the human learning capabilities. Once the encoding was implemented it only took a couple of minutes until the numbers were learned by one participant. Other participants also tried the encoding and succeeded. However, we decided to train one specialist for encoding, such that our results reflect a situation where the user is more familiar with the system.

2.3 Error correction using an EMG

In order to increase the encoding accuracy, we used an EMG. The EMG device is placed on the biceps as shown in figure 5. We tried two approaches for this. First, we programmed a threshold which compares the average measurement value during the last 150 EMG measurements to the average measurement value calculated from the EMG measurements taken 300-150 samples before the current measurement. If the average value calculated from the most recent 150 samples exceeded a threshold value which was adapted for each session, a reset command was sent to the receiver via UDP. This reset was still possible as the last encoded number is only played to the decoding participant with a delay of one number. Once our EMG broke, we had to get a new one. However, this first encoding reset condition did not work anymore, as flexing the muscle delivered the maximum obtainable EMG measurement value immediately after flexing. Hence, we decided that the resetting threshold condition was met only once an average value calculated from the last 300 sample measurements exceeded a threshold close to the maximum possible EMG measurement value obtainable. As this condition was met several times after flexing the muscle, we included a temporal condition which enabled a further reset only after one second passed since the last reset.

2.4 Machine decoding

The decoding part of the system made use of four vibrotactile motors. Originally we tried to use the

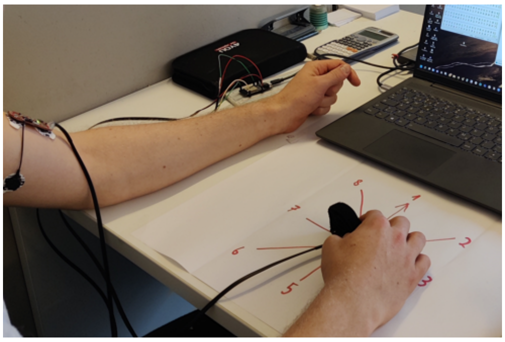

Figure 5 The EMG is placed on the arm after cleaning the skin with water or alcohol. In the figure the EMG can be seen on the left side. EMG measurements are used for correcting the previous encoded number by resetting it in case it was encoded falsely.

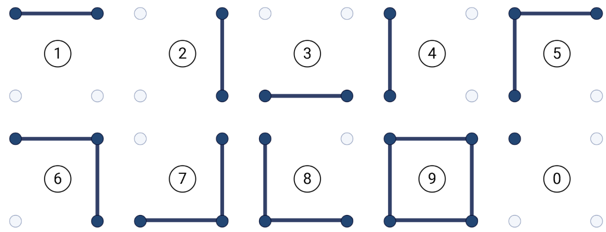

Figure 6 The first idea of decoding was to use spatial mapping by considering the four vibrators structured as edges in a square. Every two activated vibrators will form a side. Starting from the top of the square going right to all the four sides, we can map from 1 to 4. Using two sides and going right again we can map from 5 to 8. Then 9 can take the whole square and 0 can take one edge (one vibrator).

spatial dimension to make sensing more intuitive for the user. The spatial dimension idea was performed by considering the four vibrators structured as edges in a square. Every two adjacent activated vibrotactile motors will form a side as indicated in figure 6. During simulation, we found that human skin on shoulders is not sensitive enough to distinguish whether four vibrators are located closely to each other to draw the pattern. Furthermore, if the vibrotactile motors are located on the fingers, limited area will influence the feeling of subjects [6].

It was suggested that looking at the vibrotactile motors together with feeling them actually made perception easier [7, 11]. We set up an experiment to see the effect of different methods of perception on the ITR (section 2.6).

The numbers were encoded by turning the motors on or keeping them off. Continuous vibration mode is not used here. We found that participants can not detect the difference in very small vibration changes with number strings due to limited contacting area. However, differentiating between viration being turned on or off is effortless [14]. In order to overcome the skin spatial sensitivity issue, we decided to use two hands instead of only one: one hand has two vibrators and the second hand has the other two. To make the recognition easy for the perceived person, six numbers are mapped using only one hand with 2 bits. The vibrotactile motors are separated into two groups (two hands) instead of using 4-bits system as a whole, which can make the calculations complicated. This decoding strategy avoids the spatial sensitivity issue and halved the binary calculation complexity.The mapping is shown in figure 7.

In order to keep the decoding part stable, i.e decoding numbers with constant time and not make them depend on the encoding speed, and to make it display numbers in reasonable and detectable time, we added a buffer to the real-time system. It saves numbers continuously to a list, and then displays every number with a fixed delay. This delay makes it clear where one numbers ends and the next beings and also gives time to map the vibrations to numbers. The list is continuously saving numbers and the real-time system replays them ceaselessly at specific intervals, this time is then tuned for the highest transfer rate.

2.5 Full system

The full system is mainly the sum of the encoding and decoding part. In this section we'll briefly discuss the interplay between the two components. The two parts of the system were connected through a local WiFi connection based on the UDP protocol, where the encoding part acted as a client and the decoding part acted as a server. Every time the encoding part classified a number, it was immediately sent to the client. The client received this message and stored it inside a buffer. This way the encoder and decoder are not completely time-dependent on each other. For example, if numbers are more quickly encoded than decoded, these numbers will simply be stored in a buffer until they are subsequentially decoded. Similarly, if the encoding takes more time and numbers are still stored in the buffer, the decoder can simply keep on decoding.

In the case of error correction one number is always kept in the buffer. This way, it is always possible to correct the last number using an error correction signal.

2.6 Experimental setup

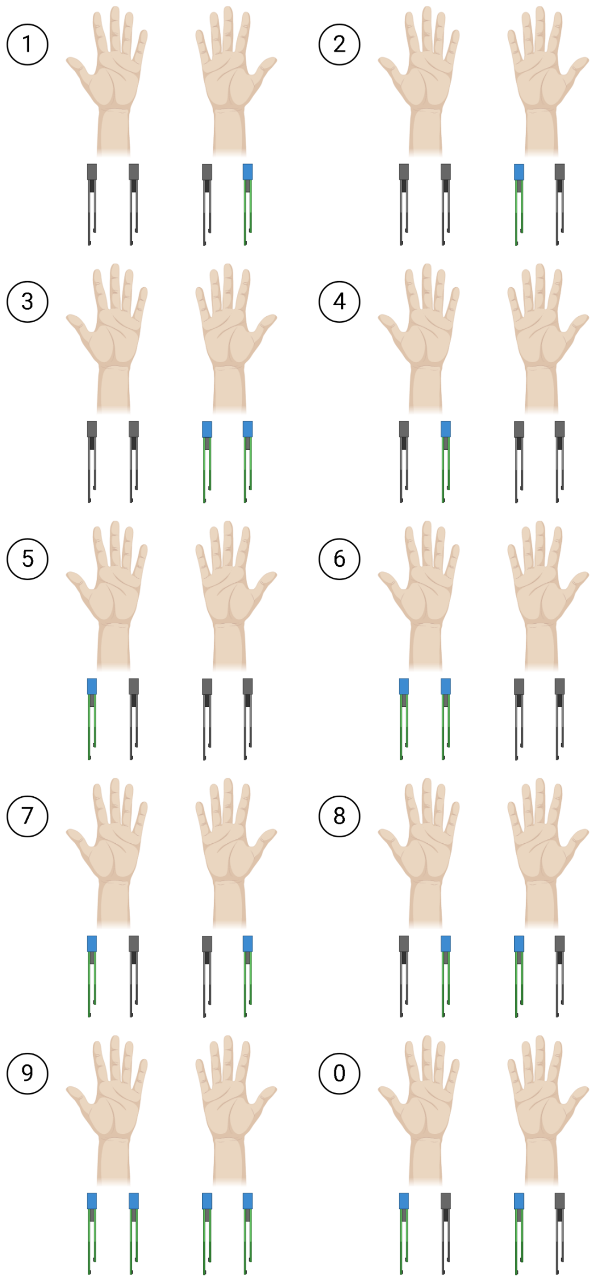

Figure 7 The numbers 0-9 are decoded in two hands. The right hand (with two vibrotactile motors) is mapped to the numbers 1-3 (in binary). The left hand is mapped to the numbers 4-6 (in binary + 3), the remaining 4 numbers (7, 8, 9, 0) are a combination of the two hands. 7 is mapped as the first combined number by taking first vibrotactile in the first hand with any vibrotactile in the second hand (we choose the second motor), 8 is mapped as the opposite to 7. Number 9 is activating all vibrotactile motors. 0 can be achieved by activating the first vibrotactile motor in both hands

Three different types of experiments were performed. One involving only the decoding part (figure 8), one involving only the encoding part of the system (figure9) and one participant, and the other involving the full system and two participants. The first experiment was performed using only the decoding part to separate its performance from potential errors by the encoding part. The second experiment was only performed with the encoding part as it features auditory feedback. This could not be used in the full system as the full system was employed in the same room and the decoder could hear this feedback and potentially use it to help with decoding.

In the first type of experiment, two types of conditions were tested. First: different time delays were performed to evaluate the optimal time delay between each displayed number in the decoding in which the user can recognise with less error and minimum time. Second: three output methods for sensory feedback, one using tactile feedback via one hand, one using tactile feedback via two hands 10, and the other using visual and auditory feedback by placing the vibrotactile motors on a table.

In the second type of experiment, three different combinations of parameters were tested: error correction, auditory feedback, and auditory input. Error correction was the same as in the first experiment type. Auditory feedback was tested to see if using audition for feedback was easier for the participant than reading it from the screen. The reasoning being that the participant has to visually focus on different lists of numbers, making it harder to focus. Using auditory feedback might decrease the load on the visual system. The same reasoning is used for the third parameter. However, instead of the feedback being auditory, the desired input is auditory. In a real-world setting this would however not be a problem as the user would come up with the input her- or himself.

In the third type of experiment, the whole system was tested for two different parameters: The system using EMG as error correction and the system without using it. In the case of error correction, the same number of numbers was given, but the participant doing the encoding was instructed to either correct wrong numbers using the EMG. Again, the ITR and accuracy were used as statistical measurements.

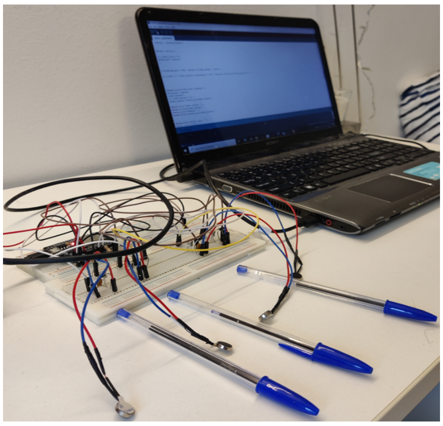

Figure 8 Decoding using auditory and visual feedback by placing the vibrators on a table. The pens were placed between vibrators such that the order of the vibrators did not change during the experiment.

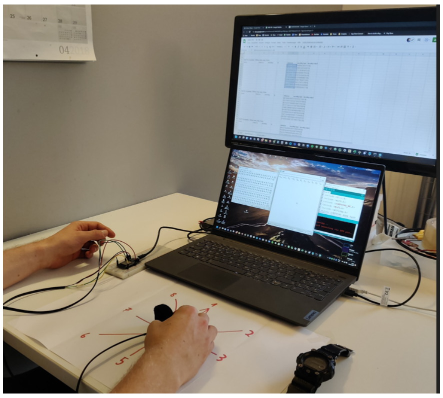

Figure 9 The participant holds the IMU in one hand for encoding. A sheet of paper containing the numbers helps with estimating the correct directions for encoding. The EMG (not visible) is placed on the left arm and can be used for resetting the last encoded number in order to increase accuracy and ITR. A comparison between EMG and non-EMG conditions can be seen in figure 14.

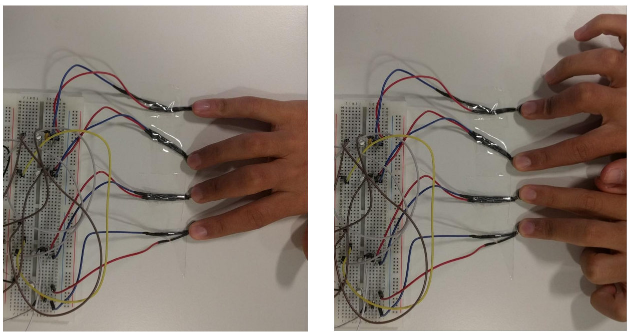

Figure 10 Decoding with one and two hands. In the experiment with one hand the decoding participant noted the numbers on paper. For the two hand experiment the decoder spoke the detected numbers out loud such that another person could note them down. A comparison can be seen in figure 7.

3 Evaluation and Results

3.1 Decoding Subsystem Evaluation

In order to evaluate the decoding part separately, and find the best setup and optimal time delay, we sent a set of 50 random numbers to the decoding subsystem and then compared these received numbers (encoding numbers) with the decoded set by the user. The important parameter to tune in the decoding subsystem is the time delay between each played number. This time is limited by the ability for the user to recognise the number before the next number is played. Ideally we aimed to find an optimal time parameter in which each number can be recognised accurately with minimum required time. To assess this, four different time delays have been tested (1000ms, 1250ms, 1500ms, 2000ms) for 3 trails each, and with two users. The result showed that the optimal time delay for the information transfer rate is 1500ms. In figure 11 we can see that increasing the time delay will give the user more time to recognise the number correctly hence a better accuracy, but since it takes more time to transmit the same amount of numbers it will at some point cause a decrease in the ITR. The optimal trade-off between time and accuracy is at the 1500ms point which gives the decoding subsystem an ITR of up to 107.2 bits per minute. In addition, we wanted to know which encoding accuracy we can achieve. Hence, we performed a 500 number encoding trial with no time constraints using EMG to correct previous numbers. Only two out of 500 numbers were encoded falsely which evaluates to an accuracy of 0.996. The whole encoding needed 995 seconds and achieved an ITR of 98.64 even though speed was not the aim of the experiment. The high ITR can be explained as the numbers were spoken out by a second person and not read from the number list. The two observed errors were encountered due to an immediate false second resetting of a number after the correct resetting of a previous mistake. Likewise, this occurred as the EMG resetting condition was exceeded a second time due to the muscle still not being fully relaxed. The resetting system could be improved by setting a temporal threshold before enabling a further number reset to higher times. The other 41 resettings of numbers were correct.

Our decoding mapping uses the idea of distributing the vibrotactile into two hands, for two reasons, one to reduce the calculation complexity into two 2-bits system instead of one 4-bits system, which makes the decoding easier by the user and two, to avoid the spatial sensitivity issue that can rise by placing the four vibrotactile motors at the same place. To evaluate this claim in the decoding subsystem, we performed a comparison between three different setups, one is decoding using vibrotactile motors on one hand, and the second is decoding using vibrotactile motors in two hands, and the last is putting the vibrotactile motors on the table and recognise numbers using visual and auditory senses. As expected, in figure 12 it can be seen that in terms of ITR, visual perception outperforms the tactile feedback, and the 2 hands setup outperforms the 1 hand setup.

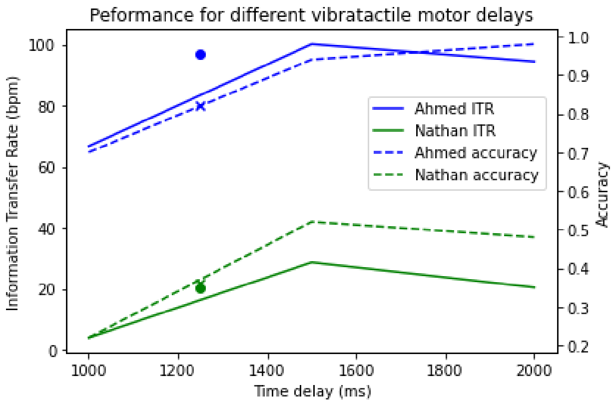

Figure 11 The evaluation of different time delays in the decoding subsystem. Time delays tested were 1000, 1250, 1500, and 2000 milliseconds. In this experiment setup, only the decoding subsystem is assessed. A set of 50 random numbers was generated and sent to the decoding subsystem, then the decoded set by the user is compared to the ground truth. Note that the 1250ms point was measured after initial analysis. Therefore it is displayed as a dot for ITR and a cross for accuracy.

3.2 Encoding subsystem Evaluation

The encoding subsystem is evaluated separately by generating a set of random numbers which the user then encoded using the system. Three different scenarios were evaluated (figure 13). One is the normal setup where the user sees the number on the screen and then encodes them after which the number is shown in another window on the screen as a way of feedback. The second scenario is listening to the number instead of seeing it - to give the user more freedom to only focus on the encoding task. The third scenario is having auditory feedback instead of the visual feedback.

3.3 Full-system Evaluation

The evaluation of the full system is done with the optimal time delay parameter in the decoding part

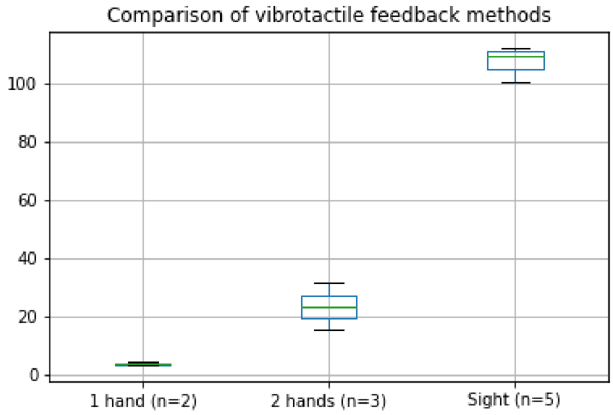

Figure 12 We evaluated the best feedback method for the user to decode the numbers. We tested three difference conditions: tactile feedback using 1 hand, tactile feedback using 2 hands, and visual/auditory feedback by placing the vibrators on a flat surface (labeled as 'sight'). The figure shows the three comparison in terms of ITR, with n as the number of trails. It can be seen that using 2 hands yields a better performance, since the vibrations are more easily separable. Using visual/auditory feedback gives the best performance.

(1500ms) and with the normal scenario in the encoding part (no speaker). The encoding part has two setups: one is with the EMG for error correction, and the other without the EMG. The first setup was used to quantitatively assess the error correction functionality. In particular, whether it will decrease or increase the ITR given the additional costs in time for correcting. Figure 14 shows that our full system can achieve an average ITR of 85.6bpm. Adding the EMG feature significantly improves the accuracy of the system, however due to the amount of time required for the error to be corrected it did not improve the overall ITR.

To assess all three parts of the system together (encoding performance, decoding performance, full-system performance) under the same conditions (no EMG, no speaker, 1500ms time delay) we performed an experiment to send as many numbers as possible within 3 minutes. We then calculated the ITR and accuracy for each part by comparing only the numbers relevant to that part (i.e. for encoding we compared the ground truth with the encoded numbers and for the full system we compared the ground truth with the decoded numbers). The results can be seen in figure 15. It can be seen that the encoding part of the system is the bottleneck, and improving the encoding would be the first step to improving the overall full-system performance.

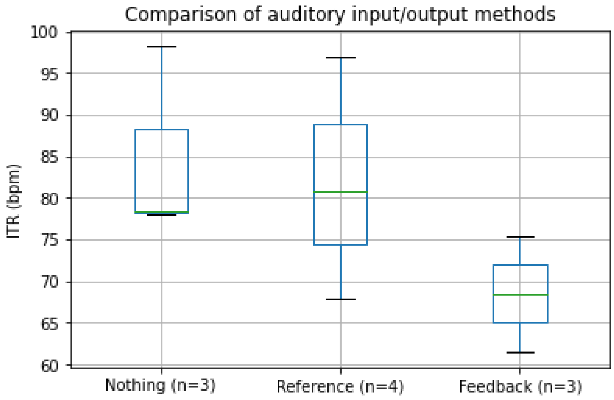

Figure 13 We evaluated the encoding subsytem performance for three different conditions: i) Having the number list and the encoded feedback as a visual input to the user. ii) Saying the numbers that should be encoded by the user. iii) Saying the numbers that were classified by the system. Interestingly, off-loading the visual system by making a part auditory did not increase the performance. A possible reason could be that using audition only disrupts the user, as the user is not in control of when the numbers are said.

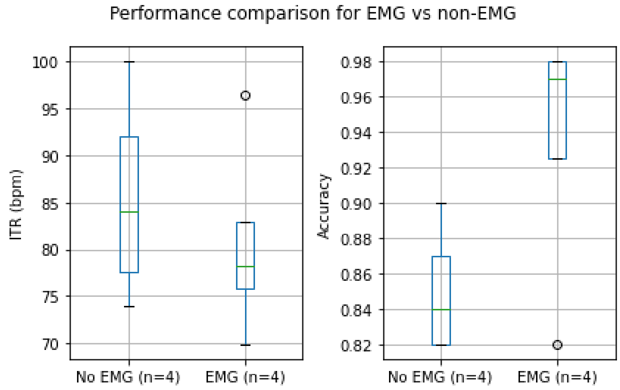

Figure 14 We evaluated the full system using the best parameters from previous experiments: 1500ms time delay for decoding and no speaker. The EMG is added to the system to evaluate the efficacy of error correction. It can be seen that the system is able to reach an average ITR of 85.6 bpm and 80.6 bpm and an average accuracy of 0.85 and 0.94 for the EMG and no EMG conditions respectively. We can thus conclude that although error correction somewhat reduces the information transmitted, it does improve the overall communication quality.

4 Discussion

The full system has achieves a high ITR during information transfer. It can decode all 10 characters and convert these to vibrations for the receiver. Visual and auditory feedback is essential for receiving messages, whereas, this also limits the conditions one needs to use the system. The room has to be quite for successful decoding and the participant should not be blind although one of both senses should also work for decoding with a slightly lower accuracy. Compared to wearable products [1], our design would restrict more on the environment. We also observed that the vibrations on the desk could influence the accuracy since it could add up to real vibrations happening in the close environment. Further design could be integrating the sensors and vibrators to some pneumatic gloves [8].

It has also been found that different parts of the human body including spatial difference would influence the accuracy of vibrotactile number detection [5]. In the decoding part, we only tested the vibration from fingertips. Some research has pointed out that feedback from the wrist might work better with different vibrators placed further apart [10].

A new study about exploiting BMI for number or letter transmission has indicated that signals in the brain can be translated by an EEG device [13]. It would be more applicable for blinds to use EEG without looking or hearing at anything. However, EEG implementation still needs a long way to go in terms of the calibration and commercial production as well as ethics [2].

Our encoding is based on several assumptions. First, we assume that a participant can learn the linear encoding fast. Second, planar movements are more reliable than 3D movements even though the angles of adjacent encoding directions is smaller. Third, a threshold method can be easily handled in terms of computational power by the Arduino. Fourth, no classifier retraining is needed for new participants.

Our system requires training the user both on the encoding and decoding end, although the amount of training is quite limited. Further limitations mainly relate to the error correction part of the system. The EMG is not always very straight-forward to set up and might require additional configuration or threshold correction before use. In addition, for error correction to work one number always needs to be kept in the buffer before being played to the encoding participant. This means that there will always be a delay of one number. Finally, the interpretability of the analysis done in this study is limited due to the low number of participants and tests done per condition.

A number of the limitations are easily addressable in future work. Changing the vibrotactile motors for a visual display for example would ease the training for decoding. The error correction could also be shown to the decoder, such that the decoder could correct for the error. This would discard the delay in numbers, how-

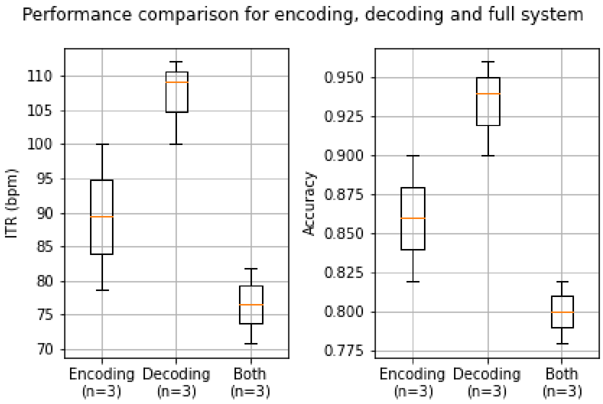

Figure 15 An overview of the different sub parts of the system can be seen in comparison to the full system. The evaluation was done under the same conditions (no EMG, no speaker, 1500ms time delay). It can be seen that the encoding part of the system is the bottleneck, both in terms of accuracy and ITR.

ever, would increase needed decoding time if many numbers have to be corrected. Error correction would be possible during natural movement (i.e. walking, moving the arm without flexing) rather than sitting or standing next to a flat surface. In addition, especially when a lot of information has to be transferred it might be insightful for the person encoding the information to have insight into the current buffer state as encoding usually has higher accuracy when the encoder has more time for encoding.

We observed that the overall ITR should be increased by changing the encoding part of the system as the encoding part seems to be the bottleneck. There are two options for this. First, improving accuracy. Second, speeding up the encoding. A further possible encoding accuracy improvements would be changing the number resetting to an IMU based reset condition as the EMG is not very reliable. In addition, this would reduce the necessity of one device and hence would reduce system cost. In addition, alternative 3D configurations could be tried despite our concerns described in the encoding section as the angle between adjacent number encoding directions would be bigger than in our approach. For improving time needed for encoding, a speaker could be included into the system which reads out the numbers to be encoded loud. This would enable the encoding person to focus on the encoding and correction process and would only require two visual focus areas (feedback output, encoding area) instead of the previous three additionally including reading the numbers on a sheet of paper or screen.

5 Conclusion

Threshold-controlled linear IMU movements and EMG signals together form the encoding part of our system. It requires little training for a new participant to use the encoding system. Our full system uses a WiFi connection for data transmission, in specific UDP is used as protocol. The transmission range is limited by the WiFi network range to approximately ten meters. This distance ensures that people receive get a signal from a distant source. The decoding part is based on binary coding which is easy to memorise. We found by analyzing multiple system trials that subjects perform better with visual and auditory information from the vibrotactile motors instead of tactile sensory detection. The maximum ITR measured during the experiments is 107.2 bpm. If focus is placed on encoding accuracy, an accuracy of more than 99.5% is achievable, even in very long encoding trials of more than 10 minutes duration. For decoding, our system is easily portable and achieves high accuracies of more than 90% on average. Our system requires no training data on both the encoding and decoding side. Further experiments for a 3D encoding scheme as well as another correction method including IMU instead of EMG should be performed to increase accuracy and ITR even further.

A Documentation

The code used for this paper can be found at:
https://github.com/scidex/imu-vibration-itr-maximization
Two scripts have to be installed on different Arduino boards. The code assumes the Arduino boards to be close enough to each other to be able to connect over a local WiFi network.
Supplementary information and data can be found here:
https://syncandshare.lrz.de/getlink/fi3AynGppD5jvn7uEdVXA3Yp/

References

[1] Miguel Reyes Adame, Jing Yu, Knut Moller, and Edgar Seemann. A wearable navigation aid for blind people using a vibrotactile information transfer system. In *2013 ICME International Conference on Complex Medical Engineering*, pages 13–18, 2013.

[2] Gordon Cheng, Stefan K. Ehrlich, Mikhail Lebedev, and Miguel A. L. Nicolelis. Neuroengineering challenges of fusing robotics and neuroscience. *Science Robotics*, 5(49):eabd1911, 2020.

[3] Seungmoon Choi and Katherine J. Kuchenbecker. Vibrotactile display: Perception, technology, and applications. *Proceedings of the IEEE*, 101(9):2093–2104, 2013.

[4] President Dr Oliver Locker-Grütjen. Brain-computer interface itr calculator.

[5] Daniel Evestedt, John Mclaughlin, and Reachin Ab. Mutual calibration of a co-located haptics device and stereoscopic display. 07 2001.

[6] Scott D Novich and David M Eagleman. Using space and time to encode vibrotactile information: toward an estimate of the skin's achievable throughput. *Experimental brain research*, 233(10):2777—2788, October 2015.

[7] Matthew S. Prewett, Linda R. Elliott, Ashley G. Walvoord, and Michael D. Coovert. A meta-analysis of vibrotactile and visual information displays for improving task performance. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1):123–132, 2012.

[8] Harshal Arun Sonar and Jamie Paik. Soft pneumatic actuator skin with piezoelectric sensors for vibrotactile feedback. *Frontiers in Robotics and AI*, 2:38, 2016.

[9] H. Christiaan Stronks, Daniel J. Parker, Janine Walker, Paulette Lieby, and Nick Barnes. The feasibility of coin motors for use in a vibrotactile display for the blind. *Artificial Organs*, 39(6):480–491, 2015.

[10] Ian R. Summers, Jon J. Whybrow, Denise A. Gratton, Peter Milnes, Brian H. Brown, and John C. Stevens. Tactile information transfer: A comparison of two stimulation sites. *The Journal of the Acoustical Society of America*, 118(4):2527–2534, 2005.

[11] Jan B.F van Erp and Marc H Verschoor. Cross-modal visual and vibrotactile tracking. *Applied Ergonomics*, 35(2):105–112, 2004.

[12] Wikipedia. Thomson problem, 2021.

[13] Francis R Willett, Donald T Avansino, Leigh R Hochberg, Jaimie M Henderson, and Krishna V Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593(7858):249—254, May 2021.

[14] Yongjae Yoo, Taekbeom Yoo, Jihyun Kong, and Seungmoon Choi. Emotional responses of tactile icons: Effects of amplitude, frequency, duration, and envelope. In *2015 IEEE World Haptics Conference (WHC)*, pages 235–240, 2015.