Proceedings of the
27th Annual Mersivity / WaterHCI Symposium
HTO Beach, June 20 - August 25, 2025

DOI (Digital Object Identifier): 10.5281/zenodo.16973160

Proceedings of the

27th Annual Mersivity / Water-HCI Symposium

"Sustainable Technology Society"

Online (www.mersivity.com)

DOI (Digital Object Identifier): 10.5281/zenodo.16973160

CONTENTS

CONFERENCE CHAIR'S REMARKS + INTRODUCTION

Steve Mann

A WARNING ABOUT TECHNOLOGY AS A PRISON

"Big Tech" and governments are together waging a war on swimming, cycling [1], and walking in the woods [2]. At the same time they're building massive highways, more cars (some self-driving), and massive data centres, creating and deploying technologies that disconnect us from nature and from each other, while building technology in such a way that we become dependent upon it, or may even be required to use it, even if we don't want to [3], [4], [5], [16].

Together with Nobel laureate Geoffrey Hinton, as well as others like Jeff Bezos, Julian Assange, Bill Gates, Elon Musk, Edward Snowden, Ray Kurzeil, and Warren Buffet, I am working hard to safeguard humanity from these and other similar existential threats, through the Lifeboat Foundation Guardian Award (https://lifeboat.com/ex/guardian.award).

This Conference Chair's Remarks + Introduction is a problem-statement of sorts, followed by an allegory to a possible solution, The Ship of Kolympi. Kolympi is just one instance and we leave it to the imagination and creativity, on the part of the reader, to generalize to other forms of cyborg technology, and technology in general.

XIR (EXTENDED INTELLIGENCE / REALITY)

Mersivity is technology that connects us with each other and our surroundings (environment), embodying XR / XI (eXtended Reality / eXtended Intelligence) which is "a form of extended intelligence and extended collective intelligence ... [to] ... help extend our sensory intelligence on land, in air, or in water ... Unlike VR which is typically a solo experience, XR is a shared experience..." [7]. It is also an umbrella term to encompass virtual reality (VR), augmented reality (AR), mediated reality or mixed reality (MR), metaverse, Internet of Things (IoT), spatial computing, and digital twin, "as the unifying framework that not only interpolates across these diverse realities but also extrapolates (extends) beyond them to create entirely new possibilities. XR is the physical spatial metaverse, bridging the physical world, the virtual world of artificial intelligence, and the social world of human interaction" [8], as illustrated in Figure 1.

XR also embodies XI (eXtended Intelligence)[7], and is therefore sometimes written XR/XI or XIR (eXtended *Intelligent Reality*), stylized as $\mathrm{X\,I\!R}$, $X\mathbb{R}$, or $X\mathbb{R}$, or simply as XI, especially when emphasizing its ability to embody AI as an extension of the physical world. Thus XI ($\mathrm{X\,I\!R}$) is the overarching interpolator and extrapolator of the "alphabet soup" of realities and intelligences as shown below:

$$\overbrace{\mathrm{X\,I\!R}}$$
AI, VR/AR/MR/DR/IR, HCI, IoT, IoT3, Metaverse, Digital Twin, ...

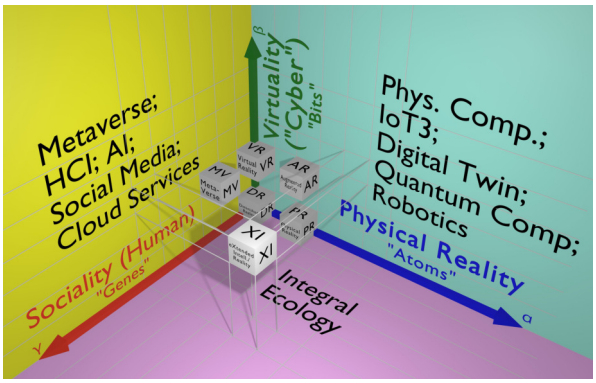


Fig. 1: **SocioCyberPhysical space** is a three-dimensional space of the social (red axis), cyber (green axis), and physical (blue axis). It shows how XIR (XI/XR) is not merely an umbrella term encompassing AI, VR, AR, Metaverse, Spatial Computing, IoT, etc., but, more importantly, it is a way to extend beyond these. (Used, with permission, from [8].)

TECHNOLOGY AT THE MOST FUNDAMENTAL LEVEL OF HUMAN EXISTENCE

I've been a "cyborg" for more than half a century, inventing, designing, building, and embodying computer vision systems that extend my reality and intelligence since my childhood, more than 50 years ago [7]. During that time, I've faced a great deal of "cyborg discrimination" and been the victim of what many in the mainstream media have referred to as the world's first cyborg hate crimes. I've not only been refused entry for wearing a computerized seeing aid, but also been physically assaulted by security guards, e.g. pushed down the stairs by security guards at the art gallery, because they were afraid my seeing aid might be recording copyrighted works of art, even though it was simply helping me see and not recording. In some sense, these attacks created a need to redesign the computer vision system so that it could also make recordings to use as evidence. Thus their fear of recording became a self-fulfilling prophecy, analogous to the "Streisand Effect [9]" where in my case, violent attacks on the right to see created a need to develop computational memory and sharing of that memory... collecting evidence.

Thus what began as an experiment in "Wearable AI [10]" expanded in scope to also became a personal safety/security system giving rise to something I call "sousveillance" ("undersight") akin to surveillance (oversight), but serving people rather than large establishments.

There are now thousands of books and research papers written on sousveillance and this work is finally being recognized as important to safeguarding humanity, e.g. I've received the Lifeboat Foundation Guardian Award for this work: https://lifeboat.com/ex/guardian2024

In the 1970s people used to walk across the street to

avoid me, but I noticed a societal shift around the 1980s when people would walk across the street to come and greet me as I'd become a bit of a celebrity of sorts, at a time when computer technology was becoming fashionable. In 1991, I brought my inventions to MIT to found the MIT Wearable Computing Project. The founding Director of the Media Lab declared that "Steve Mann is the perfect example of someone... who persisted in his vision and ended up founding a new discipline" (see articles and archive of video broadcast in http://wearcam.org/nn.htm). See Fig. 2.

The time was finally right for this technology and the ideas surrounding it to spread quickly. Even the fun playful new word I made up for it, "sousveillance" finally made it into the OED (Oxford English Dictionary).

VIRONMENTS (VESTEMENTS, VEHICLES, AND VESSELS)

The concept of Mersivity goes beyond just wearables. More generally, it includes other technologies that surround or enclose us, that extend our intelligence, or the like, such as vehicles, vessels, and vestments.

NO WAY TO "OPT OUT" OF TECH/GOVERNANCE

Technology (e.g. AI, as well as excessive governance of technology that mandates or requires it) is a Porsche with no brakes. It has become unstoppable. Our only hope now is to steer it wisely.

It is almost impossible now to avoid being recorded and we are surrounded by AI whether we like it or not. I have finally won the right to wear a computer, camera, seeing aid, etc., but society now faces a new problem. Rather than being discriminated against for using or depending on technology, we now face the prospect of being discriminated against for not using or not depending on technology. Rather than facing cyborg hate crimes, we now face the prospect of hatred or discrimination for being technology-free. We are entering a cyborg world with no way to opt-out of being a cyborg.

ONLY CYBORGS ARE ALLOWED FOOD AND WATER

I want to recount a very simple personal example. Late one night a group of us needed food and water, and there were no public drinking fountains. The only food and water source was a restaurant that was closed, except for the drive-through. When we walked up to the drive-through window to buy some food and ask for a drink of water, we were refused service because we were not in a vehicle.

Purely out of curiosity, I asked if a motorcycle or bicycle would count as a vehicle. Manfred Clynes coined the word "cyborg" (cybernetic organism) in 1960 [11] and his favorite example is a person riding a bicycle [15]. So what we are now seeing is a kind of inverse-cyborg discrimination, i.e. a discrimination against those without technological extensions of their mind and body.

On a more practical note, I told the restaurant clerk about a dumpster we were able to find that contained some old tires and scraps of metal in it, and I asked her how much of that metal and tires we needed to strap onto our bodies to be able to get a drink of water and buy some food.

I was formulating in my mind a concept of "minimum viable vehicle" as something not so much about function as about appearance, that is, not so much about becoming a cyborg, but more about looking like one, or at least completing a kind of bureaucratic "check box" for being considered a cyborg.

When dealing with this kind of inverse cyborg discrimination, we can observe a wide variety of different outcomes depending on the wide range of people, policy, and opinions at various individual establishments.

So let us pick another example which is more universal, consistent, and central to the theme of this Symposium: the most fundamental human right, the right for a non-cyborg to access water, broadly, as it pertains to government and governance.

DOWNTOWN TORONTO WATER ACCESS

Toronto is the largest city on the Great Lakes which hold 21% of the world's freshwater supply (85% of North America's). That's more than 1/5th the world's supply of freshwater. In this sense, Toronto is the world's freshwater capital and should set the best examples in terms of freshwater stewardship. Toronto has one and only one accessible beach, i.e. only one beach where a person in a wheelchair can get all the way to the water's edge, namely HTO Beach. Whereas there are claims that other beaches such as Woodbine Beach are accessible, it turns out that their wheelchair ramps only go within about 30m of the water's edge. This can be confirmed by looking at satellite images.

I have a disability and I use a specially modified wheelchair for stand-up operation (seat removed, hand piece extended for spinal support while standing upon it). My daily exercise is at HTO Beach because it is the only beach in Toronto that is accessible. We've created an outdoor classroom, lab, and outdoor university there, since I have a lab attached to the wheelchair. We collect water samples and do research on cyborg technologies that connect us to our environment (Mersivity / XI / XR). See Fig. 3.

Of all the beaches in Toronto, HTO Beach, which opened to the public 18 years ago (2007) is perhaps the most iconic in Toronto, as it is in the heart of downtown Toronto directly in front of the Rogers Skydome baseball stadium, home to the Toronto Blue Jays of Major League Baseball, situated at the base of the CN Tower. Many people who come to Toronto for the first time visit this location, as it is Toronto's most iconic, to visit the CN Tower, the Rogers Skydome, and then go for a swim at HTO Beach. See Fig 4 Although this location has a step-entry into the water, the City of Toronto refers to it as a "beach" and has also provided beach sand that can be optionally bypassed for clean (sand-free) water entry. The clean sand-free entry is helpful for accessibility. Note also

Fig. 2: Evolution of Steve Mann's wearable AI smart glass.

Fig. 3: Outdoor lab/classroom at HTO Beach, which is Toronto's only accessible beach.

the WALLadder (Wheelchair Accessible Long Ladder). These long-ladders are rare but welcome to wheelchair uses as they offer increased safety and accessibility for swimmers who are accessing the beach by wheelchair.

NO SWIMMING

Interestingly, swimming is forbidden here and elsewhere in Toronto in all but a very limited number of places: "No person shall bathe or swim in the waters of the Port and Harbour unless in an area designated for such activity as authorized by posted signs." – Harbour Master of the Port of Toronto (Fig 5). This excludes swimming in some of the safest and cleanest waters around the downtown Toronto area, including at Toronto's only accessible beach, HTO Beach, where we have been testing water quality and finding it is often clean and swimmable. This rule unfairly discriminates against persons with limited mobility by giving them no lake access. There are only a few permitted swim areas and they are all quite far from the downtown Toronto core, and none of them are wheelchair accessible.

Thus we affectionately refer to HTO Beach as TOAB (Toronto's Only Accessible Beach), i.e. the only place for persons with disabilities to swim in the lake is a place where swimming is forbidden.

Thousands of people swim in downtown Toronto at Ontario Place (designed as "the cottage for people without cottages") as well as at HTO Beach and risk receiving a fine or citation from Marine Police (See Fig 6).

Whereas Toronto ought to lead the world in making its waters accessible to persons with disabilities and to everyone, we see the opposite, where other cities like Kingston Ontario lead the way toward water access with a downtown swim area that many describe as a mini-version of HTO Beach. This area is open 24 hours a day, unsupervised (no lifeguard staff) and simply labeled "Unsupervised swimming . Swim at your own risk". See Fig 7. This swim spot in downtown Kingston opened on July 26, 2018, after hundreds of people protested the "NO SWIMMING" rule by swimming there in violation of the rule.

Around the world people are protesting against rules against swimming, while at the same time advocating for safe clean water as a basic human right. Moreover, safety itself is a basic human right, which we must protect while governments attempt to remove bicycle paths and sidewalks to put in more roads and highways [1]:

> The Ontario government has officially passed Bill 212 — a controversial piece of legislation that gives the province sweeping control over municipal bike lanes and lets construction of Highway 413 begin before Indigenous consultation or environmental assessment is complete.
>
> The fast-tracked bill, which passed at Queen's Park Monday, requires municipalities to ask the province for permission to install bike lanes ... and allows the removal of three major Toronto bike lanes on Bloor Street, Yonge Street and University Avenue ... amendment to the bill from last week, which appears to protect the government from lawsuits should someone be hurt or killed after the removal of bike lanes.

Bicycle lanes are also used by persons with disabilities, e.g. to get to Lake Ontario not just for recreation, but many with disabilities have a medical need for water for rehabilitation.

BIG-CYBORG V. LITTLE-CYBORG

The battle between motorists and cyclists is perhaps best captured by the "STREET FOR PEOPLE... CARS ARE GUESTS" meme as shown in the photo of Fig. 8. But this meme is fundamentally flawed because cars usually contain at least one person (e.g. a driver, or even if the car is self-driving, it likely contains at least one passenger). Thus it might be more correct to say that streets are for little-cyborg-tech (shoes, clothing, bicycles) and that people "wearing" big-cyborg tech (cars and trucks) are guests. This reality brings us to the concept of cyborg scalespace introduced in 2001 [3] which we shall introduce in Fig. 1 of the following paper "The Ship of Kolympi".

Fig. 4: Four photos from HTO Beach which is Toronto's only accessible beach. It is located in the center of the downtown core directly in front of the Rogers Skydome baseball stadium at the base of the CN Tower.

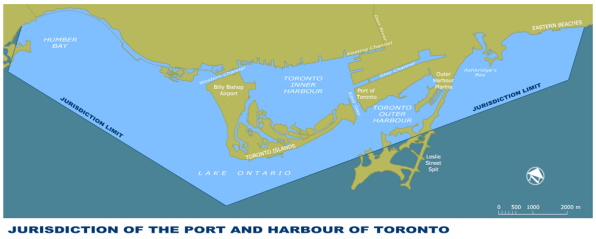


Fig. 5: "No person shall bathe or swim in the waters of the Port and Harbour unless in an area designated for such activity as authorized by posted signs." – Harbour Master of the Port of Toronto

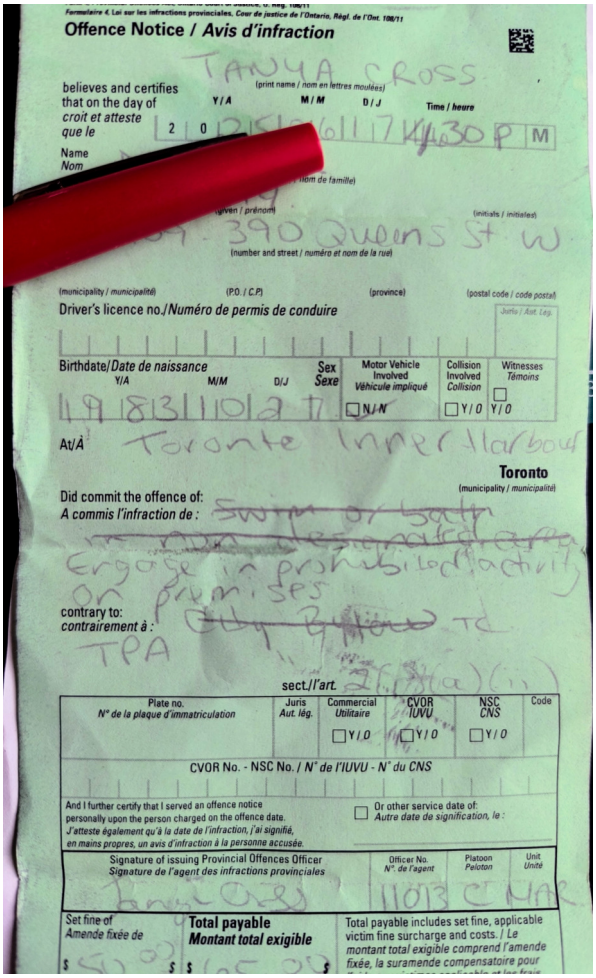


Fig. 6: Offence Notice for swimming outside one of the limited designated swim areas. It appears, though, that even the officer was confused as to the law, having crossed that out and written something about "prohibited activity on premises" as if to suggest that the lake is a "premises".

Government Bias in favour of Big-Cyborg

We must recognize an inherent conflict-of-interest. Governments fund themselves from tax dollars, and would thus tend to favor cars because they have a higher purchase price than bicycles, and cars also involve purchase of gasoline, insurance, licensing fees, and other indirect costs like increased healthcare costs associated with lack of physical activity, obesity, etc.. Understandably, governments can maximize tax revenue by discouraging swimming, walking, cycling,

Fig. 7: Gord Edgar Downie Pier at Breakwater Park in downtown Kingston Ontario is open 24 hours a day, for unsupervised swimming (no staff present) and is simply labeled "Unsupervised swiming . Swim at your own risk".

etc., which generate very little cash flow, while encouraging use of motorboats and motor vehicles which generate more cash flow (and thus more tax dollars). Indeed, Ontario's slogan / tagline changed from "Keep it Beautiful" and "Yours to Discover" to now become "Open for Business".

Fig. 8: The "STREET FOR PEOPLE... CARS ARE GUESTS" meme incorrectly assumes cars are just machines with nobody inside them, whereas in reality cars and bicycles are both examples of cyborg technology. Drivers or passengers are people too. They're just wearing a bigger and heavier vironment.

References

[1] A. Carter, "Ontario passes bill that allows major toronto bike lanes to be ripped out," *CBC News*.

[2] E. Macdonald and I. Callan, "N.s. man gets ticket on purpose for walking in woods to fight fire-safety ban," *Global News*, "August 12, 2025 5:00 am, updated August 13, 2025 10:35 am".

[3] S. Mann, "Can humans being clerks make clerks be human? *k onnen* menschen, die sich wie angestellte benehmen, angestellte zu menschlichem verhalten bewegen?" *Informationstechnik und Technische Informatik*, vol. 43, no. 2, pp. 97–106, 2001.

[4] M. Arnd-Caddigan, "Sherry turkle: Alone together: Why we expect more from technology and less from each other: Basic books, new york, 2011, 348 pp, isbn 978-0465031467 (pbk)," 2015.

[5] K. Michael, "Enslaved," *IEEE Technology and Society Magazine*, vol. 33, no. 4, pp. 5–10, 2014.

[6] S. Mann, "Can humans being machines make machines be human?" in *"Crossing the Border of Humanity: Cyborgs in Ethics, Law, and Art", International conference, Medical University of Łódz*, M. Michałowska, Ed., 2021.

[7] S. Mann and C. Wyckoff, "Extended reality, mit 4-405, massachusetts institute of technology, cambridge, massachusetts," —, 1991, also available at http://wearcam.org/xr.htm.

[8] S. Mann, M. Cooper, B. Ferren, T. M. Coughlin, and P. Travers, "Advancing technology for humanity and earth (+ water+ air)," *arXiv preprint arXiv:2501.00074*, 2024.

[9] S. C. Jansen and B. Martin, "The streisand effect and censorship backfire," *International Journal of Communication*, vol. 9, p. 16, 2015.

[10] S. Mann, L.-T. Cheng, J. Robinson, K. Sumi, T. Nishida, S. Matsushita, Ömer Faruk Özer, O. Özün, C. Öncel Tüzel, V. Atalay, A. E. Çetin, J. Anhalt, A. Smailagic, D. P. Siewiorek, F. Gemperle, D. Salber, S. Weber, J. Beck, J. Jennings, and D. A. Ross, "Wearable ai," *Intelligent Systems, IEEE*, vol. 16, no. 3, pp. 1–53, May/June 2001.

[11] M. Clynes and N. Kline, "Cyborgs and space," *Astronautics*, vol. 14, no. 9, pp. 26–27,and 74–75, Sept. 1960.

[12] M. Clynes, "personal communication," 1996.

The Ship of Kolympi: Opting out of the world's oldest cyborg technology

Steve Mann

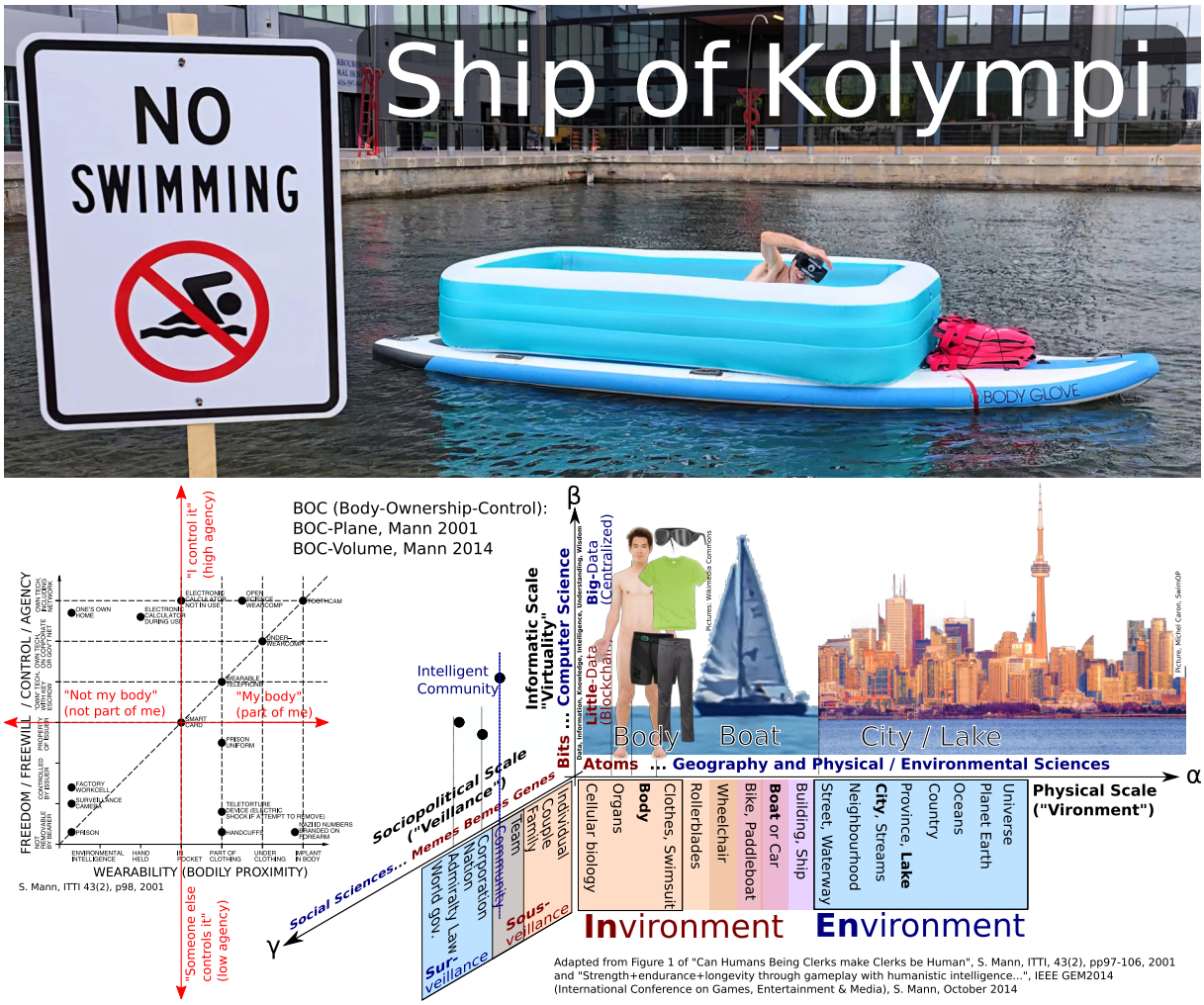


Fig. 1: The Ship of Kolympi asks questions about the right to choose not to use cyborg technology. Here an inflatable swimming pool is placed aboard a private vessel...

***Abstract*—In 1960, Manfred Clynes coined the word "cyborg" (cybernetic organism) as a nexus of humans and technology. His favorite example of cyborg technology is the bicycle. I have argued that an even better example is the boat, ship, or vessel. Indeed, the word "cybernetic" is an Ancient Greek word "kubernetes" which means "to steer" a vessel, in regards to a helmsman ("kuberman"), and forms the word "governor" in the sense of the Latin word "gubernare" meaning "to steer or govern" (as in "gubernator" or "gubernatorial" which means "of or relating to a governor"). Accordingly, "The Ship of Kolympi" is an allegory for responsible sociocyber-physical technology governance with regard to cyborg technology in its many variations, including its absence.**

Introduction

As a metaphor we might regard technology (AI, surveillance, and XR/XI cyber/cyborg-tech) as a self-driving Porsche with no brakes in the sense that it is unstoppable today. Although we can't stop it, it has a rudder and we can still steer it. We must steer it, to chart a safe and sustainable course for the future of humanity and Earth.

Cybernetics (cybersecurity, cyborgs, and everything else "cyber") comes from the Greek word kubernao, which is the root of the English words "govern", "governance", and "cybernetics". The word originally referred to steering a ship, but has now evolved into broader ideas of steering, guiding, or governing technology and society in general.

I've often been called "The world's first cyborg" in the mainstream media [13], [14], but Manfred Clynes, who coined the term "cyborg" in the 1960s often said that his favorite example of "cyborg" technology is the bicycle [15] and bicycles have been in existence for about 200 years!

Moreover, I've even argued that boats or vessels upon the water are the world's first cyborg technologies, i.e. the world's first cyborgs existed more than a million years ago, before the invention of clothing, before the invention of the wheel, and even before the existence of homo-sapiens. It was our ancient hominid (hominoid, Homininae) ancestors who were cyborgs aboard vessels more than a million years ago [16].

To stay true to the aquatic origins of all things "Cyber", Toronto serves as a good example to the rest of the world. It is the largest city on the Great Lakes. The Great Lakes hold about 1/5th of the world's (about 85 percent of North America's) freshwater supply. The world's largest lake (which holds about 1/10th the world's freshwater supply) is located in Ontario, of which Toronto is the capital.

Water is the foremost of today's concerns, from cooling data centres, to environmental issues brought on by AI.

Cybernetics + government today tends to prioritize development even if it damages the environment and our humanity. Presently, the Government of Ontario in Canada is removing bike lanes and has a ban on the installation of new bike lanes. Developers are often automobile-centric. The removal of bike lanes is intended to make more room for cars. Removal of sidewalks could be a next logical extension of this governance, to create more room for self-driving cars that carry us from our homes to supervised walking areas that might have names like the "Sunnyside Walking Pavilion" or "Woodbine Walking Station".

We're also seeing over-development of forests and beaches like the Greenbelt and Ontario Place, taking away public access to its accessible beach, and removing Toronto's only public accessibility ramps. Thanks to this development, many persons with disabilities are no longer able to access Lake Ontario.

Our political leaders are prioritizing large motorized boats like Hoverlink. The proliferation of larger and faster vessels makes swimming, paddling, and rowing more dangerous. Political leaders are clearing massive numbers of trees, e.g. downtown Toronto's only beachfront forest was cleared, under cover of darkness in one night [17], to prepare this public space for a 95-year lease to a private foreign spa company. It has even been speculated that there is an incentive for large spa-building corporations to "endangerify" or "ensewage" the lake in order to sell more pool passes [18] ("Toxic Brand"). We call this phenomenon "Denatured Water" akin to denatured alcohol (deliberate poisoning) and created some art installations, Denaturement.com as parody or allegory. We created an ice cream parlour serving denatured ice cream for display purposes only (no eating or tasting), as well as a pub with only

Fig. 2: **Endangerment as Denaturement**. Denatured whiskey, rum, etc., was created as an art installation to ask if "ensewagement" is a deliberate attempt at denaturing public waters as a means of selling more bottled water and pool passes (i.e. to keep people from drinking from or swimming in the lake) [18].

denatured spirits, wine, and beer for display purposes only, no tasting or drinking. See Fig 2.

Safety is a basic human right, as is accessibility to water by small entities like swimmers, paddlers, and rowers ("little-cyborg" tech), not just big entities like motorboats and big yachts ("big-cyborg" tech).

Efforts to clean up Lake Ontario have been progressing well, and there are now places within the inner harbour where it is safe to swim during much of time when there has not been a recent rainfall. We have finally learned that it makes more sense to just keep the lake clean than to pollute the lake raising the difficulty of purifying the water for uses like drinking. To emphasize this need, I created an art installation I called "Immersive Van Gogh" using a series of underwater Van Gogh paintings to ask the question "Would you throw garbage or sewage into an art gallery?". See Fig 3. By turning the lake into an art gallery, the Immserive Van Gogh Exhibit/Installation suggests respect for the lake. We installed also a camera looking at the paintings as a pollution monitor. See the camera and solar panel in the lower-right corner of Fig 3 and the electrical box at the bottom of the image.

We call upon governments to balance the monetary big-business pressures of development, with maintaining access to the "little business" of swimming, paddling, and rowing.

As an allegory for responsible governance, we proffer "The Ship of Kolympi" as a thought experiment about the freedom to access public space without being required to purchase, rent, or otherwise hire or possess a conveyance (vehicle, vessel, or the like).

For example, a person without a boat is often forbidden from accessing or entering water even when it is safe to do so. Similarly, a late-night pedestrian or cyclist may be denied access to food and water at a restaurant when only the drive-through window is open (as I explained in the Introduction).

Consider the fundamental human right to safe access

Fig. 3: **Immersive Van Gogh Exhibit/Installation/Pollution monitor**: the series of underwater paintings is also monitored by a camera system (see camera and solar panel in the lower right corner of the image).

Fig. 4: The Ship of Kolympi scaled down to a single pool for a single user (AI-generated figure).

Fig. 5: Modular version of The Ship of Kolympi made from an inflatable children's wading pool placed on top of a large family-sized paddleboard. This is an AI-generated image based on the photograph of Fig. 1.

to water for swimming or bathing (e.g. a cold-plunge for health, therapy, or spiritual healing). Everyone, including those with disabilities, has a basic right to safe water access, but that access is being taken away little by little, with a proliferation of "NO SWIMMING" signs, along with commercial developments of the waterfront that leave few safe places to launch a kayak, canoe, or paddleboard. Due to development, Toronto now has only one remaining beach, HTO Beach, that is wheelchair accessible all the way to the water's edge. The water quality there is quite good most of the time, but it features "NO SWIMMING" signs and Marine Police who work hard to keep people out of the water.

My thought-experiment begins with a full-size conveyance (e.g. ship or vehicle), from which pieces or parts are removed or replaced with smaller ones, until we reach a modest human-scale. It is perhaps analogous to "The Ship of Theseus" in which parts of a ship are replaced over time and the question is whether it is the same ship even after all of its parts have been replaced with new parts leaving no original parts. My question is not so much about the identity of the ship but, rather, of its mere existence.

Consider first a large cruise ship that may have several pools that can be used, even when the ship is docked at, or passing by, a "NO SWIMMING" beach. The pools could be filled with chlorinated water, or perhaps clean saltwater or freshwater from the surrounding ocean or lake, if the surrounding water were clean enough. Consider next, downsizing that ship (or building a series of ships each one smaller than the one before it) to a ship with only one pool, as shown in Fig. 4 (AI-generated figure). Here the vessel is controlled by way of a cybernetic pool that senses water-flow using underwater sensors, so that the swimmer can steer and control the vessel by swimming in the pool. Now suppose, rather than building a ship with a pool in it, we simply purchase a low-cost children's wading pool and place it on top of a family-size paddleboard, such as a 7-person Bodyglove Crusader, as shown in the photograph of Fig 1 and as illustrated in the AI-generated image of Fig 5. In the setup shown in the photograph of Fig 1, there are two SQ34 hydrophones in the pool, each receiving water wake from the swimmer in the pool

Fig. 6: Swimming in my own private pool aboard my own private vessel. Presently the pool contains more than 320 million cubic miles (more than 1300 cubic kilometers) of water. For safety, the vessel is tied to a rope so that it cannot go more than about 30 feet (about 10m) from shore.

and each being amplified and fed to a thruster on the paddleboard. The port-side thruster is responsive to an output of the port-side hydrophone and the starboard side thruster is responsive to an output fo the starboard-side hydrophone. So when the swimmer swims to the right, the ship steers to the right, and when the swimmer swims to the left, the ship steers to the left. In this way the ship is controlled by swimming in its pool. Thus the swimmer has the feeling of swimming in the lake, but is not guilty of swimming in a NO SWIMMING area, any more than swimmers aboard a cruise ship are guilty of swimming when using the pools aboard the cruise ship.

In the prototype of Fig 1, the means of propulsion is by way of human-powered electric thrusters powered from an array of hydrophones that generate electricity from swimming in the pool. Thus the "ship" is propelled forward by swimming in its pool. Here the pool water is totally separate from the lake.

Now suppose that pool were to be directly connected to the surrounding water by a pipe of ever-increasing size. Eventually, as the ship gets smaller, and the pipe connecting the ship's pool to the lake gets larger, we might approach something that resembles the Intex Relaxation Island Float Raft, which is a 7-person vessel, topologically equivalent to a torus (swim ring), as shown in Fig. 6. No lifejackets are legally required as the vessel is not underway. Lifejackets should always be worn for safety when aboard a vessel that is underway, but it is difficult to swim wearing a lifejacket, so it is common and safe to remove the lifejacket to jump into a pool that is aboard a vessel. For safety the vessel is tied to shore. Because mooring is not allowed, the other end of the rope is tied to my modified wheelchair which is legally considered part of my body. Thus I am swimming in a pool aboard my own private vessel that is tied by a rope to a part of my body that is on shore. The pools is bottomless and its water is the lake's water which is connected to the world's oceans, and therefore the amount of water in my pool is the water of the Great Lakes and the oceans of the world. In this sense we have more than

Fig. 7: State-of-Float™ research aboard the floating island which is also used for cold plunges and cold-plunge research.

320 million cubic miles (more than 1300 million cubic kilometers) of water in this pool.

We also use the vessel for research on what we call State-of-Float™ , e.g. mindfulness and meditation using the InteraXon Muse brain-sensing headband (Fig. 7), as well as for cold-plunges in its on-board pool. See Tzanetakis et al. in this Symposium Proceedings.

It is not hard to imagine that vessel decrease in size to a 5-person, 4-person, 3-person, 2-person, then 1-person swim island like a 1-person tire inner-tube or swim ring.

Alternatively, consider an inflatable rowboat with the bottom cut out of as shown in Fig. 8 and in the video at https://www.youtube.com/watch?v=8GBwDQHAHo8 showing the construction process (e.g. unboxing of the new vessel, inflating it, and cutting out the bottom), and getting it underway. For safety, I carry two lifejackets aboard the vessel, one for myself, and an additional lifejacket in case I end up needing to rescue someone that I might find in the water. (I have rescued people from the water on previous occasions.). See Fig. 9.

Next consider a paddleboard, propelled by human power, at first with full-size paddles, e.g. a standup paddleboard (SUP). It can be paddled, or it can be

Fig. 8: The Towboat™ or Rowfloat™ , a version of towfloat that is a boat.

Fig. 9: I'm not swimming, I'm just towing my vessel (with its built-in pool) out into the lake. Safety First! The Towboat/Rowfloat is equipped with two lifejackets, two whistles, towropes, and communiations and navigation equipment, wearable navigation system, headup display for navigation, signalling, and other safety equipment.

towed with a front-tether. For safety, it is beter to waist-tether a paddleboard than to ankle-tether it, because in an emergency it is easier to free one's self from the waist tether (e.g. entanglement in rapids). See Fig 10.

Fig. 10: I'm not swimming, I'm just towing my stand-up paddleboard (SUP) out into the lake. For safety there are personal floatation devices, emergency food and water, camping supplies, signaling, communications equipment, lighting, and emergency lighting equipment aboard the vessel. The board itself is brightly-colored and highly visible to match the bright orange "high-viz" swimcap. There are also paddles aboard the vessel.

Next, consider a paddleboard that uses hand-paddles. Hand-paddles were invented by Benjamin Franklin at the age of 11, for swimming in Boston's Charles River. They are often used by swimmers for developing arm strength.

Next, consider a paddleboard propelled by someone without paddles at all, e.g. just hand-paddling with bare hands. Paddleboarding with bare hands (i.e. without a paddle) is known as "prone paddleboarding" or "traditional paddleboarding" in contrast to modern (stand-up) paddleboarding. Traditional paddleboarding is pictured in the 1781 engraving by ship artist John Webber who accompanied Captain James Cook to the Sandwich Islands in 1778. See the excerpt of the engraving showing a surf-style of paddling with a paddleboard in the foreground of Fig. 11.

Next might come a smaller and smaller paddleboard, surf rescue board, surfboard, or the like. Some surfboards are inflatable such as shown in Fig 12, and although one can't stand on such a surfboard very effectively, it is excellent for practicing of paddling skills that are important to surfing. It folds up into a small pocket-sized package that weighs less than 1 pound (less than 1/2kg).

As we continue to shrink the Ship of Kolympi, we might perhaps eventually end up with a small kickboard, propelled in much the same way as the Bodyglove Crusader prototype, by kicking. Surfing shortboards about 6 feet (about 2m) long are common, as are bodyboards about 3 feet (about 1m) long. Such vessels facilitate strokes very similar to swimming, e.g. paddling "freestyle" or paddling "butterfly" or paddling "breaststroke". See for example Fig. 13

WEARABOAT™

Consider now a wearable paddleboard, even smaller, like perhaps 1 foot long (about 30cm long). See Fig 14. The experience is like paddling on a surfboard that one can never "wipe out" from.

NECKBOAT

Whereas yachts are usually measured in feet, going smaller, we might want to start measuring vessels in

Fig. 11: Indigenous Polynesian paddleboarding in the 1700s (traditional paddleboarding with no paddle) visible in the foreground near the bottom of the picture.

Fig. 12: Paddleboarding on a pocket-sized inflatable surfboard that weighs less than 1 pound (less than 1kg).

Fig. 13: Not swimming, just paddleboarding. A short styrofoam bodyboard approx. 3 feet (approx. 1m) long is perhaps a minimum viable vessel (minimum allowable cyborg prosthesis) that one must posess to be allowed to enter the water.

inches, such as the 4-inch (approx. 10cm) vessel of Fig 15. Eventually we come to realize the legal fiction of a "NO SWIMMING" sign.

We often do crossings from downtown Toronto (mainland) to the Toronto Island. For safety, we always go in large groups, accompanied by large vessels moving slowly with us.

Interestingly, one of our members of SwimOP.com (Swim at Ontario Place) joined a group of us to do a crossing from Ontario Place East Island to the Toronto Islands, landing on Hanlan's beach at Centre Island wearing a Neckboat. During the crossing he wore the Neckboat pictured in Fig. 16.

Interestingly he lost this vessel (it came loose from the string around his neck) on his return trip. As he did not become aware of losing the vessel until landing back on East Island, he must have been swimming for the portion of the trip during which the vessel came loose from his neck. In this way he may have unknowingly broken the NO SWIMMING rule. However, because he started at Hanlan's beach which is a designated swimming beach, where swimming is allowed, one could argue that he wasn't in violation of the rule, as there appears to be no specification as to how far one can stray from a designated swimming spot before being considered to be no longer at that beach.

Fig. 14: "Wearaboat™" (wearable boat) is a miniature paddleboard, attached to the body so it doesn't fall off.

A CONTINUUM OF SMALLER AND SMALLER VESSELS

The laws regarding swimming fail to define what is meant by "vessel", leaving us to wonder just how much cyborg technology one must "wear" in order to be allowed access to the water, e.g. especially for those without means or time to travel to a place where swimming is allowed, or those with disabilities who have no choice but to swim where they can access the water.

We seek not so much to answer all these questions but simply to raise questions. As James Baldwin once said "the purpose of art is to lay bare the questions that have been hidden by the answers".

Consider a continuum of smaller and smaller vessels (Fig. 17). At what point is one committing the crime of swimming as one chooses vessels of less and less encumberance?

MICROYACHTS, NANOYACHTS... YOKTOYACHT™

The concepts of microyachts and nanoyachts have been introduced. Now I introduce the YoktoYacht, which might be defined as the smallest yacht for a person to go "NOT SWIMMING". Yokto is the metric prefix of 10^{-24}.

We might wish to adopt a sense of creative humour in defining what might be a soft-boundary between swimming and NOT SWIMMING, perhaps at the edges of human visibility and eyesight.

I made a series of variously-sized "YoktoYachts" (wooden boats) using a laser cutter/engraver, as shown in Fig. 18

It was found that a 3/4 inch (approx. 19mm) long boat was the smallest upon which legible text could be written, using the available laser-engraving equipment, to identify it as NOT SWIMMING. See Fig 19

Having spent more than a year at Stanford University, and studying some of the work done there, I am presently exploring the use of a scanning tunneling microscope (STM) to arrange individual atoms or molecules in a two-dimensional pattern to make text identifying a very small vessel as "NOT SWIMMING". I name this vessel the QuectoYacht™ , after the smallest metric prefix "quecto" which means 10^{-30}.

Unfortunately, for the swimmer, this might require that Marine Police equip themselves with a scanning electron microscope to read the indicia on the yacht, so that they can determine whether a person is guilty of swimming, or is merely boating (e.g. paddleboarding).

YOKTOYACHT MEME

As a form of outreach, we ran a series of boatbuilding workshops and engaged members of the community, as well as other members of our research teams to make boats as well as to personalize them as works of art (Fig. 20).

SAFELY CONNECTING TO NATURE

An official might be tempted to pick a point along the continuum in which the boater can stay dry as being a defining boundary of what one might think of as a real vessel. However, another question we can address is how to make an extremely safe vessel that also allows the boater to completely connect with the water. If outriggers are added to a very small 12-inch (about 30cm) wide paddleboard, it becomes stable enough to support one or more tall masts (Fig. 21). The vessel has a visibility mast that can be seen by other vessels and can also house a computer vision system, camera, radar, and navigation system with antenna, etc., feeding into a VR (Virtual Reality) or XR (eXtended Reality) headset.

This vessel can also accommodate illuminated masts, 10 feet (appprox. 3m) tall one on the starboard pontoon, and another on the port pontoon, for visibility, and can support three full-size flags one flying from each mast, so that the vessel can be clearly seen by other vessels from at least a mile or so (approx. 1.6km) away. An example flag mast is shown in Fig 22.

At the same time, the captain of this ship can do freestyle, butterfly, breaststroke, and backstroke paddling and remain completely in the water, soaking wet, and even see underwater with an underwater VR / XR headset connected to the onboard sonar system for navigation that works above and below the water, combining radar, sonar, and computer vision for maximum safety and navigational capability.

CONCLUSION: SAFETY AS A HUMAN-RIGHT

In summary, safety is a basic human right, as is safe access to safe water. We call upon all authorities to prioritize safe water accessibility and to ensure that water remains safe for everyone, not just those in big boats.

Fig. 15: “Neckboat™”, a necklace-based vessel... perhaps a good-luck charm for protection from the Marine Police.

Fig. 16: Neckboat worn by one of the members of a group of people crossing from Ontario Place East Island to the Toronto Islands, pictured here after landing on Hanlan’s beach at Centre Island.

Fig. 17: Variously sized embodiments of the Ship of Kolympi randing from 11feet (approx. 3m) to 1 inch (approx. 3cm).

We envision a place we call “Mobase” as a base-of-operations for mobility and accessibility in downtown Toronto around Peter Street Basin (“Mobasin”), HTO Urban Beach, and Harbourfront Canoe and Kayak Centre. And we also call upon ourselves and each other to engage in Safetymaking™, to make the world safer, especially when and where our authorities can’t

Fig. 18: Yoktoyachts™ lasercut/engraved in varying sizes to determine the smallest size with legible text.

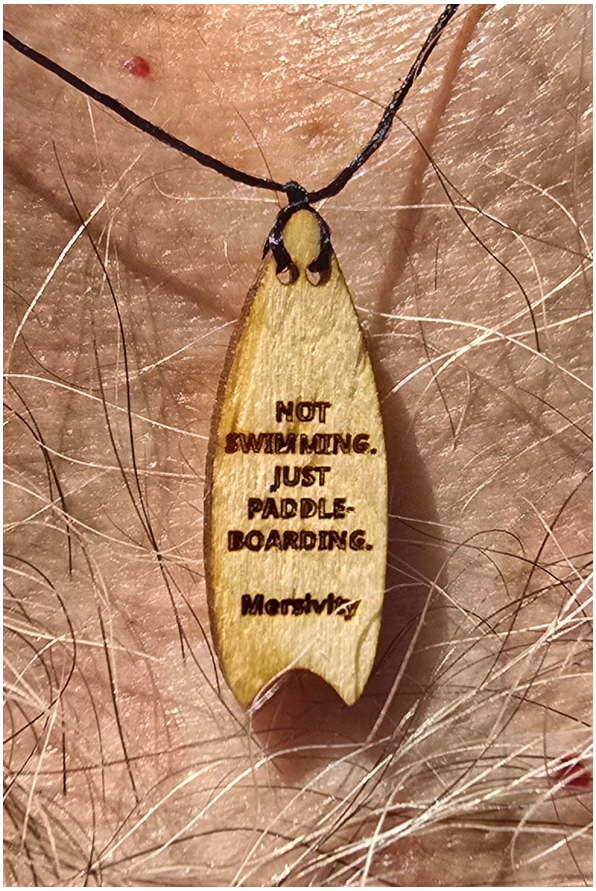


Fig. 19: A boat approximately 3/4 of an inch (approx. 19mm) long was found to be the smallest boat upon which clearly legible text could be written on the rough wooden surface, using the available laser-engraving equipment. We are presently exploring the use of a scanning tuneling microscope to arrange individual atoms on a metal surface to create the QuectoYacht™.

or won't. For example, when the safety ladders began to fall off the dock wall, as the nuts and bolts were coming loose, we contacted all of the various authorities to request that they be repaired or maintained. Over a period of many years, our calls were ignored and the ladders began to deteriorate, and some were lost as they fell off the wall and sank to the bottom of the lake. As a result, we formed Safetymaking™ and purchased the necessary supplies to maintain them (Fig. 23). Safetymaking is the community-driven process of transforming dangerous public spaces into safe accessible places that foster connection between people and their environment for health and well-being. This process begins through raising awareness and contacting other organizations who might be able to help. Safetymaking also builds trust and community-connectedness, in finding solutions to make the community safer when larger, distant, or more bureaucratic organizations can't or won't. Safetymaking follows the concepts of placemaking when authors like Jane Jacobs campaigned for cities to be more pedestrian-friendly rather than only car or commerce-friendly. Her idea of "eyes on the street" can be embodied through the modern concept of safeguarding humanity through sousveillance.

More generally, Safetymaking aims to balance big boats, big data, and big-watching (surveillance) with little boats (swimming or paddling), little data (blockchain) and little-watching (sousveillance). We're all cyborgs in the Manfred Clynes sense of a person riding a bicycle or a person riding on a vessel. But

Fig. 20: Community involvement in the YokoYacht meme... boatbuilding as art.

Fig. 21: Even a very narrow 12-inch (approx. 30cm) wide paddleboard can be made stable enough to support one or more tall masts, simply by adding outriggers. Here two outriggers convert the miniature paddleboard to a trimarin that can support up to 3 tall masts for visibility and safety.

we must never adopt technologies that become compulsory or enslave us. We must balance the benefits of technology with the right to, at least once in a while, not use the technology. The right to swim is the right to be a non-cyborg, even if only briefly.

The Swimboat™, Wearaboat™, Neckboat™, "Swoating™", Ship of Kolympi™, etc., were invented by S. Mann in Canada, and also form what is known as a Minimum Allowable Cyborg Technology (MACT), Minimum Allowable Vessel (MAV) or Minimum Viable Vessel (MVV); see Mann, Steve. "... Liminal Spaces of Body, Ownership, and Control." In Humanity In-Between and Beyond, Monika Michałowska, ed, pp. 141-152.: Springer International Publishing, 2023 [19]. There is an inherent conflict-of-interest that makes it profitable for governments and corporations to pollute our lakes and rivers, creating "denatured waters", making it unsafe to swim, so people need to use motorboats rather than risk falling from a smaller vessel like a kayak or paddleboard, and so that people need to buy bottled water to drink rather than tap water from the lake. The MACT

Fig. 22: Freestyle (front-crawl) “Swoating™ ” in police-infested waters. A Marine Police boat, visible in the background, has just passed by and did not have issue with this activity. A highly visible flagpole also serves as a lifejacket rack.

Fig. 23: Safetymaking™ involves first reaching out and raising awareness, and then if that doesn’t work, creating a community-driven process of transforming dangerous public spaces into safe accessible places that foster connection between people and their environment for health and well-being, not just for cars, commerce, and large gasoline-powered boats. Here a safety ladder is being repaired after notifying authorities and waiting many years to no avail.

is a first-step towards raising awareness about this potential conflict-of-interest. Getting more people “in” the water is the first step toward protecting our supply of drinking water and the MACT/Kolympi can help with this goal.

See also “The Ship of Theseus”, also known as “Theseus’s Paradox”.

Notes: The ship’s name, Kolympi or Kolymvisis, derives from the Greek word for ”swimming”, e.g. no swimming; όχι κολύμπι; óchi kolýmpi swimming; κολύμπι; kolýmpi the ship of swimming; το πλοίο της κολύμβησης; to ploío tis kolýmvisis

Acknowledgements

I wish to thank the thousands of members of SwimOP who helped keep this effort going, as well as many of the core members, Patrick, Nuno, Perry, Dan, Dari, Alex, and Mike Dopsa, who helped with the Oculus Quest 3 for safe navigation, as well as Despina who helped with the boat paintings. I also thank Bill Mann for getting me started on boat-building since early childhood.

References

[13] P. Nowak, “The world’s first ’cyborg,’ Steve Mann, says always being connected to others can make people see the world in a different — and better — light,” *CBC News*, Monday Dec. 22, 2003.

[14] S. M. (with Hal Niedzviecki), *Cyborg: Digital Destiny and Human Possibility in the Age of the Wearable Computer*. Randomhouse (Doubleday), November 6 2001, iSBN: 0-385-65825-7.
[15] M. Clynes, "personal communication," 1996.
[16] S. Mann, "Can humans being machines make machines be human?" in *"Crossing the Border of Humanity: Cyborgs in Ethics, Law, and Art", International conference, Medical University of Łódz*, M. Michałowska, Ed., 2021.
[17] B. Robertson, "Ontario place bulldozed under cover of darkness and people are livid," *blogTO*, October 3, 2024.
[18] S. Mann, "Ontario place spa will ruin toronto's most spectacular beach, which is used all year," *Toronto Star*, April 28, 2023.
[19] M. Michałowska, *Humanity in-between and Beyond*. Springer, 2023, vol. 16.

Neurophysiological Effects of Cold Water Exposure: Working Towards Improving Personal Brain Health and Cognition Through EEG-Based Analysis of Brain Waves

Despina Eleni Tzanetakis
ECE516 Mersivity.com
University of Toronto
Toronto, Canada

Christina Mann
ECE516 Mersivity.com
University of Toronto
Toronto, Canada

Ahmet Sencer Aksu
ECE516 Mersivity.com
University of Toronto
Toronto, Canada

Somin Mindy Lee
ECE516 Mersivity.com
University of Toronto
Toronto, Canada

***Abstract*—This study investigates the neurophysiological effects of cold water exposure by analyzing EEG data collected using a MUSE headband. The research evaluates changes in brainwave activity before, during, and after cold plunges, focusing on the modulation of alpha, beta, delta, theta, and gamma bands. Results demonstrate significant increases in alpha, beta, and gamma activity, suggesting enhanced focus, relaxation, and cognitive function following cold exposure. The study also explores the implications for long-term brain health and cognitive resilience.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Cold water immersion, or cold plunging, has gained attention for its potential physical and psychological benefits, including reduced inflammation, improved circulation, and enhanced mood [1]. Recent studies have begun to explore the neurophysiological mechanisms underlying these effects, yet few have quantitatively examined how cold water exposure alters brain activity in real-time. With the rise of consumer-accessible electroencephalography (EEG) technology, it is now possible to investigate these effects through non-invasive brainwave monitoring.

Brainwave activity can be categorized into several frequency bands—delta, theta, alpha, beta, and gamma—each associated with distinct cognitive and emotional states. For example, alpha waves are commonly linked with relaxation and mental clarity, while beta and gamma waves are associated with alertness, focus, and high-level cognitive processing. Understanding how these brainwave patterns change in response to cold exposure can offer new insights into the neural basis of stress resilience, mood regulation, and cognitive enhancement.

This study utilizes the MUSE™ headband, a portable EEG device, to track and analyze changes in brainwave activity before, during, and after cold water immersion. The goal is to determine how cold exposure modulates brain function both acutely and over short-term recovery periods. By analyzing EEG data across key brain regions and frequency bands, this research contributes to a growing body of work aimed at improving personal brain health and cognitive performance through accessible, real-time neural monitoring.

II. Background

The human brain operates through complex electrical activity that can be measured non-invasively using electroencephalography (EEG). EEG captures voltage fluctuations generated by neuronal activity and classifies them into distinct frequency bands: delta (0–4 Hz), theta (4–8 Hz), alpha (8–12 Hz), beta (12–40 Hz), and gamma (above 40 Hz). Each of these bands corresponds to different cognitive and physiological states. Delta waves are associated with deep sleep and restoration, theta with creativity and light sleep, alpha with calm and relaxed alertness, beta with focused mental activity, and gamma with high-level cognitive functions such as perception, memory integration, and consciousness [2].

Recent interest has grown around using EEG to explore how external environmental factors, such as temperature and sensory stimulation, influence brain activity and overall cognitive performance. One such factor is cold water immersion, a practice used in athletic recovery and increasingly adopted in wellness communities for its reported psychological benefits [1]. Cold exposure activates the sympathetic nervous system and triggers a stress response, but it is followed by parasympathetic rebound, which can restore physiological balance and promote mental clarity [3, 4].

Alpha waves, in particular, are considered a neural marker of relaxation and readiness. A key metric used to evaluate alpha-related brain function is Peak Alpha Frequency (PAF), which represents the dominant oscillation within the alpha band. PAF has been associated with cognitive performance, attention, and mental well-being. Higher PAF values are correlated with better working memory and faster information processing, while lower PAF is linked with depression, attention-deficit/hyperactivity disorder (ADHD), and cognitive decline [5, 6].

In the context of cold exposure, changes in brainwave dynamics, especially in the alpha, beta, and gamma ranges, can

offer insight into the brain's adaptive response to physiological stress and recovery. Temporary decreases in PAF may occur during cold-induced stress, followed by rebounds that enhance cognitive and emotional states [6]. Long-term exposure has been hypothesized to promote neuroplastic changes, contributing to improved emotional resilience and stress tolerance [7].

Consumer-grade EEG devices like the MUSE™ headband have made it possible to monitor brain activity in real time outside of traditional lab settings (Fig.1). Although these systems don't offer the same spatial resolution as clinical-grade EEGs, they still deliver reliable temporal data that's well-suited for tracking broader patterns in brainwave activity. Thanks to their portability and user-friendly design, tools like the MUSE are well-suited for studying how everyday interventions, such as cold plunging, can influence cognitive and emotional states [9].

Fig. 1. MUSE S Headband used in the study

This study builds on existing research by quantifying the neurophysiological impact of cold water immersion using EEG, with the aim of understanding how brainwave activity is altered in the moments and hours following a plunge. By investigating these effects across different frequency bands and brain regions, the research contributes to a deeper understanding of how environmental stimuli influence cognitive health and brain function.

III. Methods

This section outlines the experimental setup, EEG data acquisition using the MUSE™ headband, and the full signal processing pipeline used to analyze brainwave activity in response to cold water immersion. The study also includes a control task (medium-level Sudoku) to establish a cognitive baseline, allowing comparative analysis of brainwave activity before, during, and after cold exposure.

A. EEG Acquisition System

EEG data was collected using the MUSE™ 2 headband, a consumer-grade, portable electroencephalography (EEG) device designed for real-time monitoring of brain activity. The MUSE system includes four dry electrodes positioned at AF7, AF8, TP9, and TP10, corresponding to prefrontal and temporal-parietal regions. These locations are associated with emotional processing, sensory integration, cognitive attention, and relaxation [2].

To facilitate wireless data streaming from the MUSE to a computer, we used Open Sound Control (OSC) in conjunction with an ESP32 microcontroller, programmed to receive data via Wi-Fi and forward it through a serial port to **CoolTerm**, a terminal application for logging real-time sensor data. Data was stored as a plain text file containing raw data readings across all four EEG channels.

B. Experimental Protocol

Participants were instructed to undergo a cold plunge in three phases:

1) Before Cold Exposure: Participants completed a 5-minute baseline recording while solving a medium-level Sudoku puzzle to simulate a focused cognitive state.
2) During Cold Exposure: EEG data was recorded as participants immersed themselves in cold water (5°C) for 10 minutes (Fig. 2) (Fig. 3) (Fig. 4).
3) After Cold Exposure: EEG was recorded again immediately after and three hours later, allowing analysis of both short-term and sustained neurophysiological effects.

Fig. 2. Temperature of water with thermometer reading

C. Signal Processing Pipeline

EEG data collected from the MUSE headband underwent a structured signal processing workflow to extract meaningful frequency-domain features. The key stages of the pipeline are as follows:

1) Data Preprocessing:

- Raw EEG data (a 4-channel time series in `.txt` format) was loaded into a Python environment.
- Invalid or corrupt rows were removed.
- Data was converted into a 32-bit floating-point NumPy array for numerical stability and precision.

Fig. 3. Subject 1 partaking in cold plunge

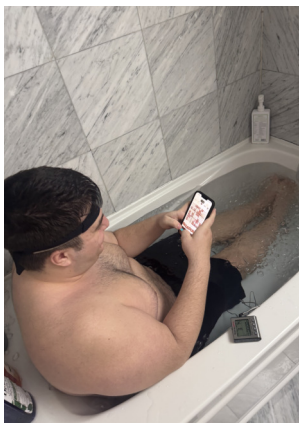


Fig. 4. Subject 2 partaking in cold plunge

2) Bandpass Filtering: A Butterworth bandpass filter (5th order) was applied to limit the frequency range from 0.5 Hz to 50 Hz. The filtering was implemented in zero-phase mode using `scipy.signal.filtfilt` to preserve temporal alignment and avoid phase distortion.

3) Sliding Window Segmentation: The filtered signal was segmented using 5-second windows with a 2-second overlap to balance temporal resolution and frequency stability.

4) Power Spectral Density (PSD) Estimation: Two methods were used for estimating PSD:

- Fast Fourier Transform (FFT): Converted time-domain segments into frequency-domain spectra.
- Multitaper Method: Applied multiple tapers to each segment before FFT to reduce spectral leakage. Implemented via `psd_array_multitaper` from the MNE-Python library, this method averages power estimates across tapers for a more robust PSD.

5) Bandpower Extraction: Power was integrated across standard EEG frequency bands:

- Delta (0.5–4 Hz): Deep sleep and recovery
- Theta (4–8 Hz): Memory, creativity
- Alpha (8–13 Hz): Relaxation, focus
- Beta (13–30 Hz): Alertness, cognition
- Gamma (30–50 Hz): Learning, perception

6) Logarithmic Transformation: To improve comparability across participants and conditions, power values were converted to decibels (dB) using the formula:

$$\text{Power}_{\text{dB}} = 10 \cdot \log_{10}(\text{Bandpower})$$

7) Output and Visualization:

- A `.csv` file was generated with columns representing sample number and bandpower (in dB) per channel and per frequency band.
- Line plots and bar charts were created to visualize trends in bandpower over time and across different cold plunge phases.
- Comparative plots were generated to assess changes relative to the Sudoku control baseline.

This methodology enabled the extraction of reliable and interpretable neural biomarkers for assessing the cognitive impact of cold exposure. The combination of wearable EEG hardware, microcontroller-based streaming, and signal processing techniques provided the overall system for conducting this neurophysiological experiment.

IV. Results

The plots shown in Fig. 5 and Fig. 6 demonstrate the processed EEG data from two subjects for before, during, and after the cold plunge.

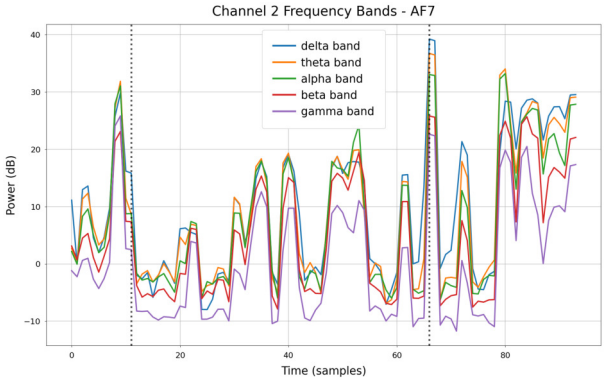


Fig. 5. Brain wave data from Despina's cold plunge. The left black vertical line shows the entry into the bathtub and the right one shows exit time from the bathtub

A. Cold-Water Immersion

During the cold-water immersion, all EEG frequency bands exhibited an initial spike in power followed by a gradual decrease, indicating an immediate neural response to the thermal stress.

B. Post-Exposure Observations

After the subjects exited the cold environment, a sustained increase in EEG band power was observed relative to their resting state, suggesting a rebound or recovery effect.

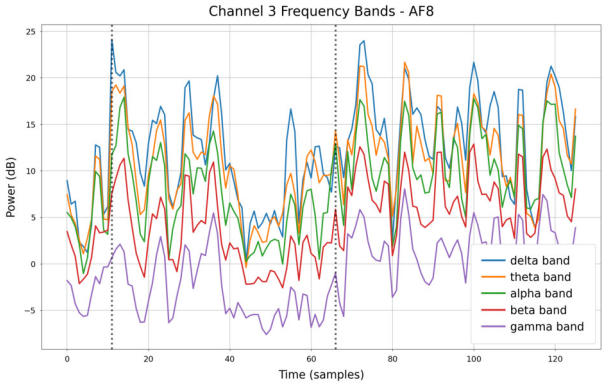


Fig. 6. Brain wave data from Christina's cold plunge. The left black vertical line shows the entry into the bathtub and the right one shows exit time from the bathtub

C. Meditation Session Outcomes

A 3.5-minute meditation session was conducted for a subject at around 3:00 pm on a day when no cold water immersion occured (Fig. 7) and on another day at around the same time for the same subject 3 hours after the cold-water exposure (Fig. 8). The analysis revealed:

- Beta and Gamma Bands: Significantly heightened power readings.
- Alpha, Delta, and Theta Bands: Slightly elevated power readings.

The data shows that cold-water immersion triggers a sharp, immediate increase in EEG band power, which gradually tapers off as the exposure continues. After the plunge, brainwave activity remained elevated across all frequency bands compared to the initial resting state. Interestingly, a follow-up meditation session appeared to amplify this effect, especially in the beta and gamma ranges. These patterns offer a clearer picture of how the brain responds to and recovers from short-term thermal stress.

V. Discussion

This study examined the neurophysiological effects of cold water immersion on brainwave activity using EEG data collected from multiple participants. The key findings support the hypothesis that cold plunging modulates brain function in measurable ways across several EEG frequency bands, with especially notable changes in the alpha, beta, and gamma ranges. These changes were most evident immediately after cold exposure and continued for hours afterward, indicating both acute and sustained cognitive and physiological effects.

A. Alpha, Beta, and Gamma Activity After Cold Exposure

Following cold immersion, participants exhibited an immediate and consistent increase in alpha band activity, which is commonly associated with relaxed alertness, mental calm, and focused attention. This rise in alpha power suggests that cold plunging may initiate a parasympathetic rebound following the initial stress response, allowing the brain to shift into a restorative state [5].

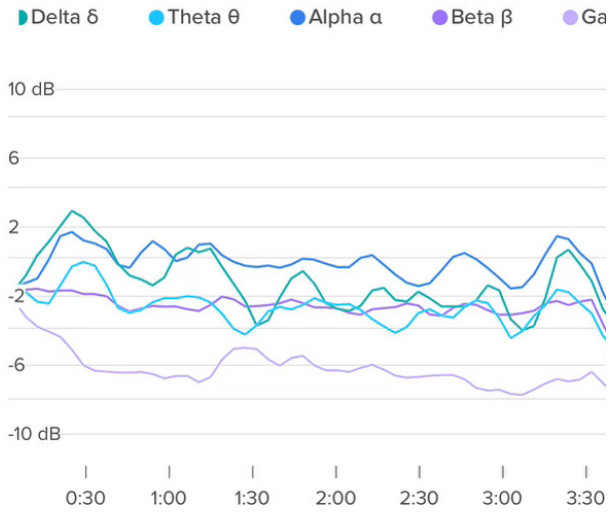


Fig. 7. Brain wave data from meditation without doing a cold plunge. All bands show a baseline value

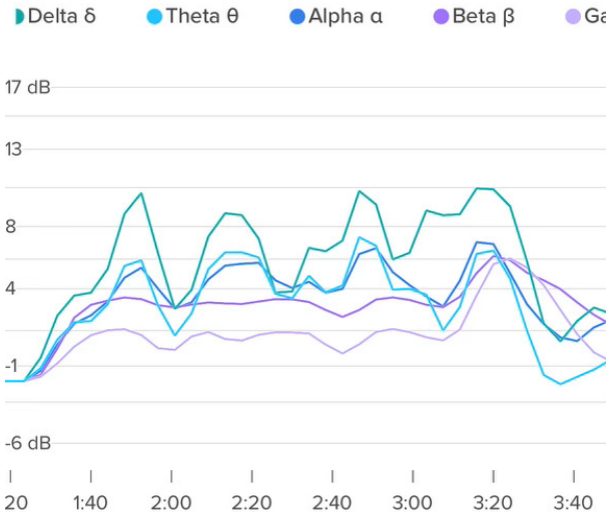


Fig. 8. Brain wave data from meditation without doing a cold plunge. All bands show an elevated result compared to the baseline without the cold plunge

Simultaneously, beta and gamma band activity, linked to active cognitive processing and sensory integration, increased after the cold plunge and remained elevated in recordings taken three hours later. These increases imply enhanced mental clarity, attention, and perhaps an improved readiness for learning and task engagement.

B. Peak Alpha Frequency (PAF) as a Neural Biomarker

A particularly insightful metric in this study is the analysis of PAF, defined as the specific frequency within the alpha band (typically 8–13 Hz) that exhibits the maximum power. PAF is a well-established biomarker for cognitive performance, emotional state, and neurological health [6].

During cold exposure, participants exhibited a transient drop in PAF, likely due to sympathetic nervous system activation and stress. However, in the post-immersion recordings, PAF values rebounded and, in some individuals, exceeded baseline levels by 0.5–1.5 Hz. This rebound may reflect increased parasympathetic activity, enhanced neuroplasticity, or improved cortical synchronization, all of which are associated with better attention, memory, and mood regulation.

C. Age and Peak Alpha Frequency Trends

PAF is known to decline with age, typically peaking during adolescence and young adulthood, followed by a gradual decline in older adults. Fig. 9 shows a clear inverse correlation between participant age and baseline PAF, consistent with existing literature [5].

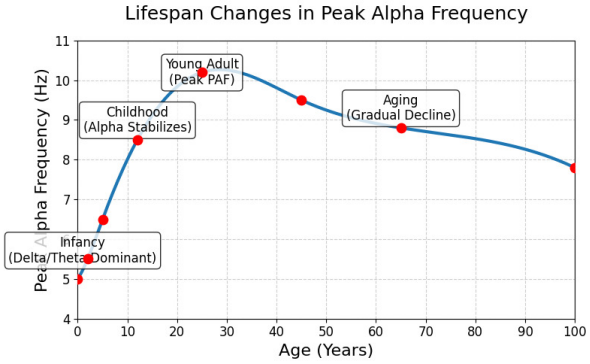


Fig. 9. Peak Alpha Frequency (PAF) vs. Participant Age. A general decline in baseline PAF with increasing age is observed. Post-plunge recovery was more pronounced in younger individuals.

D. Control Task Comparison: Sudoku vs. Cold Exposure

The inclusion of a medium-difficulty Sudoku puzzle as a control task allowed us to compare the EEG effects of mental engagement versus physiological stress and recovery. While Sudoku elicited modest increases in beta activity (consistent with task-related concentration), it did not significantly affect alpha or gamma bands. In contrast, cold plunging had a more holistic effect across multiple bands, suggesting a unique neurophysiological signature that combines relaxation with heightened awareness, something not typically achieved through cognitive tasks alone.

E. Implications

These results open several promising avenues for future research. First, cold exposure may serve as a low-cost, accessible method for enhancing cognitive readiness and stress resilience. Second, PAF could be used as a dynamic biomarker to assess and track individual responses to cold therapy or other wellness interventions over time.

VI. Future Work

In the next phase of our research, we aim to integrate advanced machine learning techniques to improve the accuracy and interpretability of EEG signal analysis. A promising direction involves the use of convolutional neural networks (CNNs), which are particularly effective at identifying spatial and temporal patterns in time-series data such as EEG recordings. By transforming raw EEG signals into spectrograms or other time-frequency representations (Fig. 10), CNNs can autonomously learn discriminative features associated with cold-water immersion and its cognitive or physiological effects. This approach has the potential to outperform traditional statistical methods in classifying or predicting brain states.

We also plan to investigate recurrent neural networks (RNNs)—with a focus on Long Short-Term Memory (LSTM) architectures—to capture temporal dependencies in EEG data. LSTMs are well-suited for modeling sequences and may help uncover how brain-wave dynamics evolve before, during, and after cold-water exposure. With properly labeled datasets, an LSTM model could detect subtle transitions in neural activity and enable real-time feedback for interventions or wellness applications.

Another crucial component of our future work is enhancing data quality through artifact removal and data augmentation. We intend to implement preprocessing pipelines using techniques like Independent Component Analysis (ICA) to filter out non-neural signals (e.g., muscle movements or eye blinks), ensuring that only clean data is used for model training. Additionally, we aim to incorporate transfer learning strategies, allowing us to leverage existing EEG datasets from similar domains and mitigate limitations posed by small sample sizes.

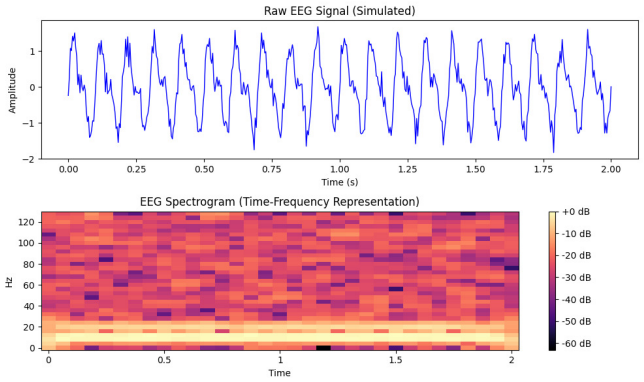


Fig. 10. Spectrogram of EEG simulation data

References

[1] Cleveland Clinic, "What to know about cold plunges," *Cleveland Clinic Health Essentials*, Mar. 22, 2023. [Online]. Available: https://health.clevelandclinic.org/what-to-know-about-cold-plunges

[2] P. L. Nunez and R. Srinivasan, *Electric Fields of the Brain: The Neurophysics of EEG*. Oxford, U.K.: Oxford Univ. Press, 2006.

[3] M. J. Tipton, "The initial responses to cold-water immersion in man," *Clin. Sci.*, vol. 77, no. 6, pp. 581–588, 1989.

[4] M. Kox, V. Stoffels, J. A. M. Smeekens, et al., "Voluntary activation of the sympathetic nervous system and attenuation of the innate immune response in humans," *Proc. Natl. Acad. Sci. USA*, vol. 111, no. 20, pp. 7379–7384, 2014.
[5] W. Klimesch, "EEG alpha and theta oscillations reflect cognitive and memory performance: a review and analysis," *Brain Res. Rev.*, vol. 29, no. 2-3, pp. 169–195, 1999.
[6] E. Angelakis, J. F. Lubar, and S. Stathopoulou, "Electroencephalographic peak alpha frequency correlates of cognitive traits," *Neurosci. Lett.*, vol. 371, no. 1, pp. 60–63, 2004.
[7] P. Carrive, "The periaqueductal gray and defensive behavior: Functional representation and neuronal organization," *Behav. Brain Res.*, vol. 182, no. 2, pp. 215–226, 2006.
[8] G. Schlaug, A. Norton, K. Overy, and E. Winner, "Training-induced neuroplasticity in young children," *Ann. N. Y. Acad. Sci.*, vol. 1169, no. 1, pp. 205–208, 2009.
[9] O. E. Krigolson, C. R. Williams, B. Norton, C. D. Hassall, and J. L. Colino, "Choosing MUSE: Validation of a low-cost, portable EEG system for ERP research," *Front. Neurosci.*, vol. 11, art. no. 109, pp. 1–10, 2017.

MuseCroc Mobile - Highly Accessible EEG, PPG, and fNIRS Data Collection & Visualization

1st Mitchell Seitz
MannLab
Toronto, Canada
mseitz@mseitz.dev

2nd Nishant Kumar
MannLab
Toronto, Canada
nishant.kumar@mail.utoronto.ca

3rd Homey Poon
MannLab
Toronto, Canada
pnhomey@gmail.com

4th Rocklen Jeong
MannLab
Toronto, Canada
rocklen.jeong@mail.utoronto.ca

5th Aydin Hosseingholizadeh
MannLab
Toronto, Canada
aidin.py3@gmail.com

6th Steve Mann
MannLab
Toronto, Canada
mann@eyetap.org

***Abstract*—The Muse S Athena is a low-cost, portable EEG (ElectroEncephaloGraphy), PPG(PhotoPlethysomoGraphy), & fNIRS (fuctional Near Infrared Spectroscopy) consumer device released in 2025. Related earlier models of Muse hardware have used in research settings. The goal of this study is to evaluate and demonstrate the effectiveness of a mobile application in recording EEG & fNIRS data from the human brain directly to a .csv file, with the goal of facilitating further research into brain biometrics in situations where traditional equipment may be out of reach or impractical to use.**

I. Introduction

Fig. 1: Muse Croc Logo

MuseCroc Mobile is an application intended to provide easy, free access to biometric data in the form of EEG (ElectroEncephaloGram), PPG (PhotoPlethysomoGraphy, and fNIRS (functional Near InfraRed Spectroscopy) data collected by the Muse headbands. MuseCroc Mobile is written in Java for Android, and uses the Muse SDK (Software Development Kit) to facilitate headband connection. The program consists of a foreground service, which wraps the Muse SDK to automatically handle headband connection and settings and is intended to allow re-use of program code in new projects, as well as a UI that interacts with the service to allow users to connect to Muse headbands, collect, export, and store data, and view real-time connection quality, brain activity, and artifacts.

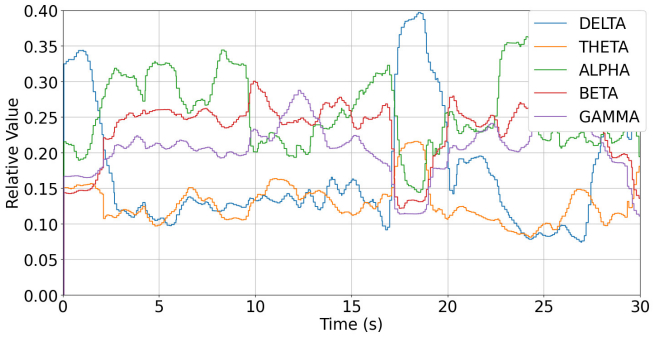


Fig. 2: Avg Relative Frequency Values

II. Background

A. EEG - ElectroEncephaloGraphy & The Muse Headband

EEG (ElectroEncephaloGraphy) is a technology used for measuring electrical activity originating in the human brain, with its first use recorded over a hundred years ago [1]. EEG is a common and safe method of observing the activity of the human brain, with application in the treatment and research of various health conditions including seizures, tumors, brain injury, sleep disorders, Alzheimer's, and other neurological disorders [2].

The Muse is a commercially available EEG headset, used for both wellness and research purposes. Some examples of research conducted with the Muse EEG include assessment of well being indicators [3], sleep studies [4], Event Related Potentials (ERP) research [5], and assessment of the effectiveness of Muse-Assisted mindfulness training [6].

Additionally, the low cost, ease of use, and accessibility of the Muse hardware has resulted in its use as a tool for large scale EEG experiments, including those on audiences [7] and it has been validated as a low-cost, mobile method of assessment of cognitive fatigue [8].

The Muse devices can collect EEG data from the electrodes TP9, AF7, AF8, and TP10 [9]. As found during the development of the MuseCroc Mobile, the Muse can output EEG data in the form of Raw EEG in μV, Relative Frequency values as proportion of total EEG signal, or Absolute Frequency values in μV.

B. Functional Near Infrared Spectroscopy (fNIRS)

The Muse S Athena is the latest Muse headband offering from Interaxon, released in 2025. One of the biggest improvements offered by the Athena is the new functional near-infrared spectroscopy (fNIRS) system.

fNIRS is a technology used to measure Hemoglobin and Deoxyhemoglobin in the human brain, with examples of its use including assessment of task-related cortical function and monitoring of cerebral oxygenation in neonatal ICUs [10]. fNIRS systems operate by measuring the absorption of relevant wavelengths of light, allowing operators to measure blood flow and hemoglobin species in the brain [10].

The Muse S Athena is a recently released device that allows for EEG, PPG, and fNIRS Data collection. The Muse S Athena is a compact headband that can be worn while sleeping or engaging in activities. This allows for a plethora of exciting new research opportunities - low-cost, location-independent, and accessible fNIRS in addition to EEG for research in diverse environments and settings.

C. PPG - Heart Rate Tracking

PhotoPlethysomoGraphy, or PPG, is a low-cost & non-invasive method of measuring blood volume changes in the microvascular bed of tissue that is commonly used to evaluate the function of the cardiovascular system, including heart rate [11]. Multiple models of Muse headset contain PPG sensors, and collecting this data allows for the evaluation of heart rate over time.

D. Gyroscope and Accelerometer

In addition to EEG, PPG, & fNIRS, Muses are also capable of collecting accelerometer and gyroscope data. This data has been used in the physical tracking of participant's movement [9]. The ability to track movement and orientation alongside EEG, PPG, and fNIRS data with a portable system can allow for diverse potential research applications, many of which could exist outside of typical lab or medical settings.

E. What is a "MuseCroc"?

"MuseCroc" is a term coined at MannLab, Toronto, Canada, and refers to a device or software that captures data broadcast by a Muse device [9], [12]. There have been two previous MuseCroc projects - the more recent being a ESP32-Based device that can capture raw Muse data and write it to an SD card [9], and an earlier ESP32-based device used to control interactive devices with brain activity [12].

The intention of this project is to carry the spirit of the MuseCroc to accessible software to allow for easy access to Muse data. No source code from the first two projects is present in this application, however the spirit and goal remain.

F. Motivation

Conventional EEG and fNIRS devices continue to be costly, stationary, and limited to laboratory or clinical environments. These limitations circumscribe the variety of populations and contexts that can be investigated. Mobile and consumer-class devices, by contrast, promise inexpensive, scalable, and location-independent acquisition. The aspiration of MuseCroc Mobile is thus not merely to be a proof of technical feasibility, but to open up biosignal acquisition to students, independent researchers, and small laboratories, allowing them to conduct substantive neuroscientific inquiry without access to specialized infrastructure. This motivation is in accordance with recent directions in mobile health, edge computing, and citizen science, in which small, portable instruments are facilitating greater inclusivity in scientific investigation.

III. Methodology

The primary goal of this development project is to create an easy to use, portable, and freely distributable application to support and streamline accessible research using Muse headbands. The secondary goal is to create a reusable wrapper for the Muse SDK in order to simplify SDK use and allow for developers and researchers in MannLab and collaborating organizations to create custom mobile applications for use in research.

In order to support these goals, the architecture of the program has been broadly divided into two parts - the Muse Foreground Service, which interacts with the Muse SDK and allows for easy portability between programs, and the main UI, which exists as an easy to understand interface for observing real-time muse data and file management.

The Muse Foreground Service handles the detection and connection to Muse headbands, and contains buffers to hold streamed data as well as file recording components that manage the collection of streamed data to .csv files. The Muse streams data via packets over BLE, and each time a packet is received the relevant buffers are updated and written to the .csv file. The foreground service has a much simpler interface than the SDK as a whole, and contains methods and variables that allow for automatic management of device settings and data collection.

A. Program Use & Operation

The user interface of the program consists of four sections.

The first section contains a list of available Muse headbands, and will display information about the currently connected device if present. This section is visible in Figure 4.

The second section contains live visualization of EEG electrode connection quality, as well as the average relative frequency values across all electrodes. This section is intended to allow for live assessment of EEG fit and data streaming when a Muse is connected. This section is visible in Figure 5.

The third section contains a selection of data types that can be collected and recorded by the Muse. The Muse Foreground Service will automatically select ideal presets and listeners on the Muse device based on desired data (EEG

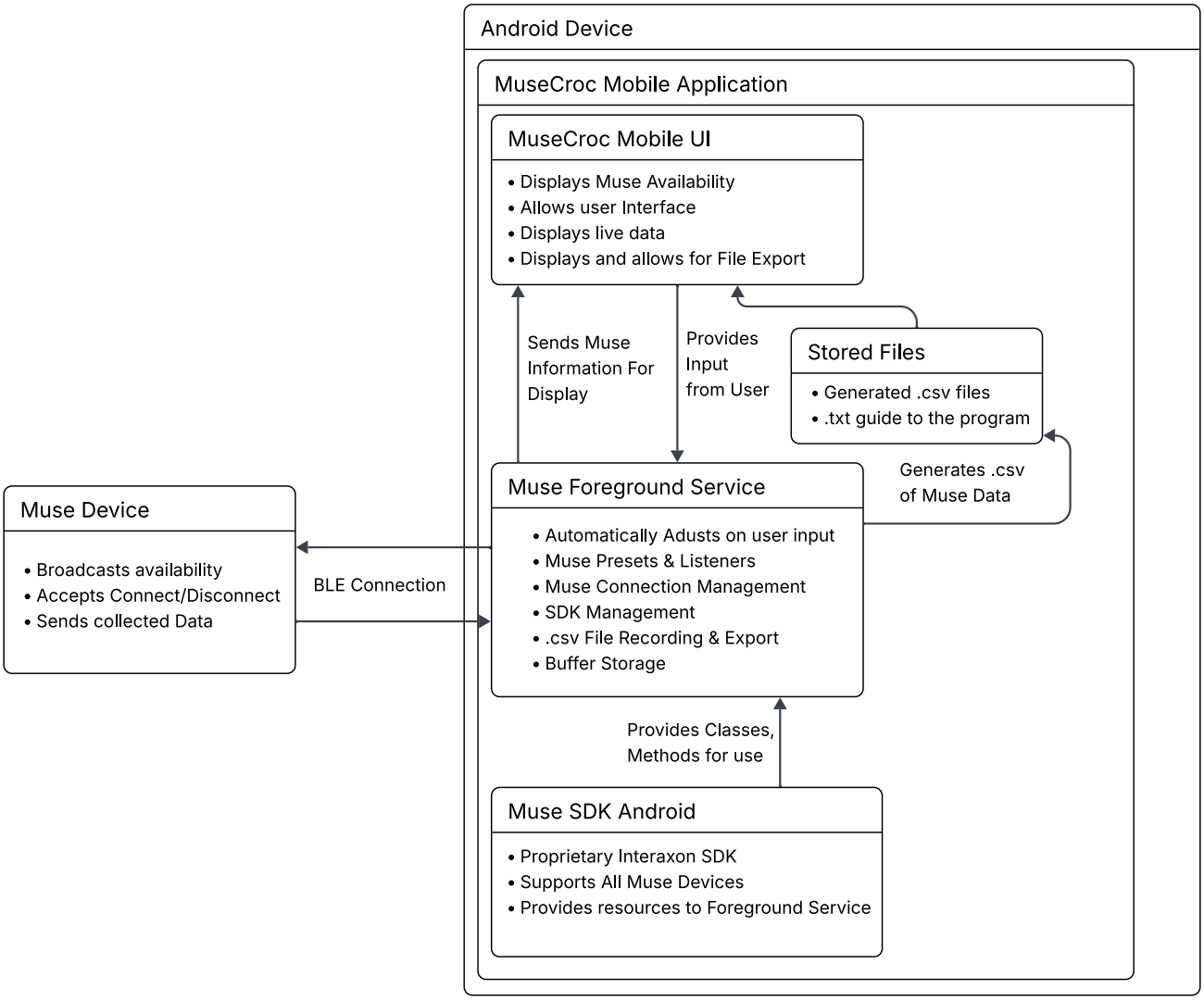


Fig. 3: Program Architecture

Raw, Absolute/Relative Frequencies, PPG/fNIRS, Accelerometer/Gyroscope). Finally, the bottom of the program contains the recording toggle, which allows the user to start and stop recordings, and a menu to choose recorded files to export from. It also contains a guide that can be exported to allow for understanding of the data in the .csv file and program operation. This section is visible in Figure 6.

The Final program is intended to be distributed as a .apk, with source code re-used in MannLab in order to enable future research and development.

IV. Results

To demonstrate data collected with the Muse S Athena and MuseCroc Mobile, I created a 30-Second recording of Raw EEG, Relative Frequency values, and PPG/fNIRS data collected with a Muse S Athena. The resulting .csv file was exported and graphed to create the visualizations below.

A. Artifact Indication and Connection Quality

The Muse outputs live indications of Artifact detection along each electrode, as well as ”HSI”, which is used to represent the quality of the electrode connection as determined by the Muse.

The graphs presented in Figure 7 represent the artifact detection as indicated by the Muse device’s internal methods by electrode, with green regions signifying artifact-free connection.

Observation of the graphs generated from our trial data tells us that connections along TP9, AF7, and AF8 are considered mostly artifact-free by this particular metric provided by the Muse SDK, while TP10 seems to have a higher incidence of disruptive artifacts during the test. This asymmetry is consistent with the expected variability that occurs due to electrode placement and contact pressure, highlighting the importance of monitoring per electrode in mobile context.

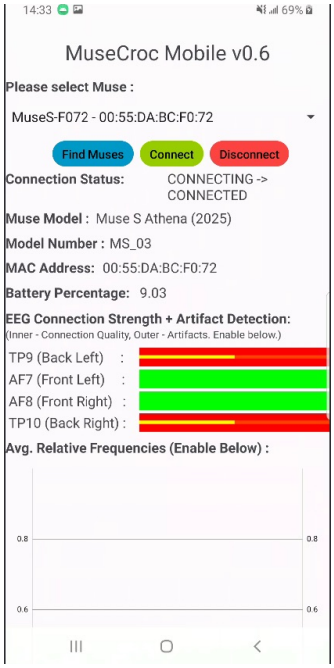


Fig. 4: Screenshot of Muse Information & Connection Control

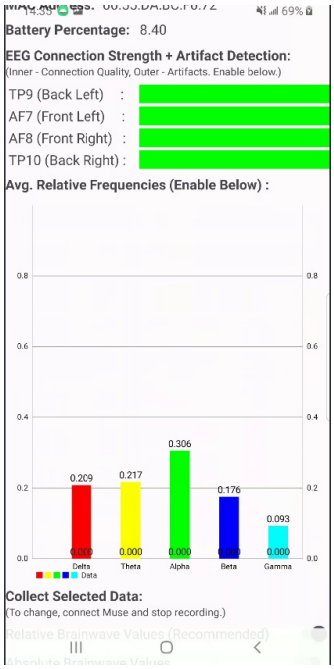


Fig. 5: Screenshot of EEG Connection Quality & Relative Frequencies

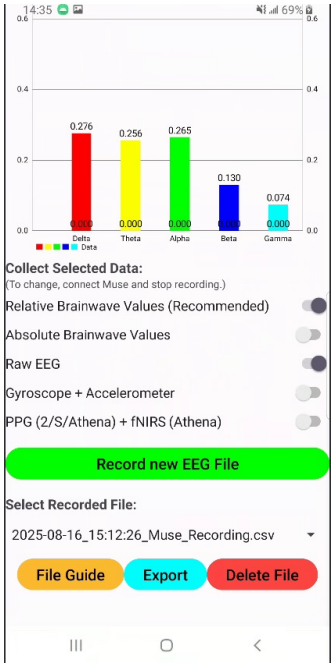


Fig. 6: Screenshot of Streaming Toggles & File Export Controls

In addition to artifact detection, we can view the HSI status by electrode as well. These values are generated by the Muse and sent alongside EEG data, and are intended to serve as an indicator of the quality of connection on the provided electrodes. These values are explained in the SDK, with a value of 1 meaning a good connection exists, 2 meaning a poor connection exists, and 4 meaning no connection or a very poor connection exists. The HSI values over our 30-second experiment are visible in Figure 8.

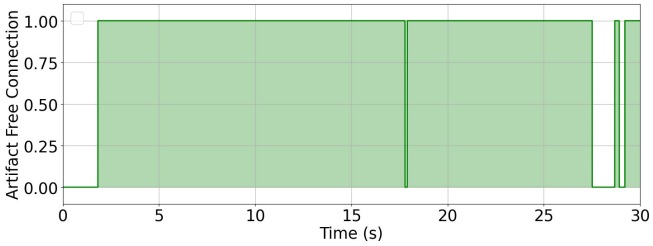
These HSI and Artifact Detection values are the same as what is displayed to the user in real-time using the horizontal bar-shaped Connection Strength/Artifact Detection indicators in the UI. Although researchers may wish to use different methods to evaluate signal quality and artifact presence, these metrics remain useful because of their simplicity and real-time generation by the Muse.

B. Raw EEG

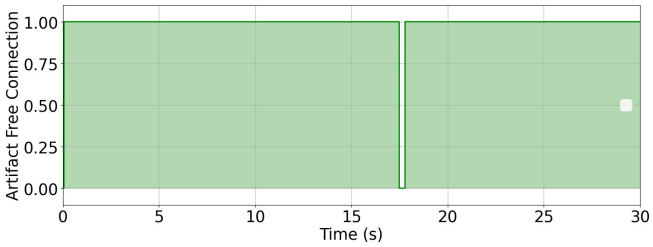
The next metric we will be displaying is the Raw EEG data in microvolts, which is collected by the Muse along each of the TP9, AF7, AF8, and TP10 electrodes. The graphs of these values in uV over our 30 second experiment are visible in Figure 9.

C. EEG Relative Frequencies

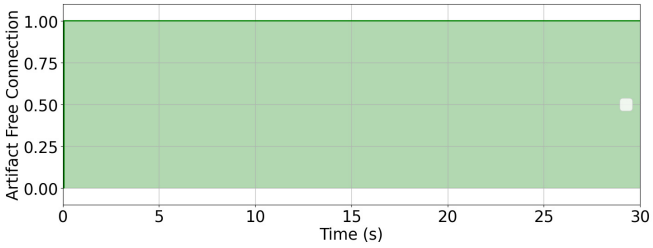
The Muse SDK allows for live output relative and absolute band power of different frequency ranges. Much like Artifact Indication and Connection Quality, researchers may wish to use different methods to generate these values. However, these packets provide an easy to use and simple to visualize source of insight into brain activity. These values are created by the



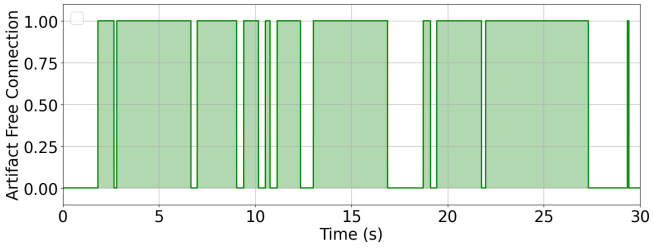
(a) TP9 Artifact Detection



(b) AF7 Artifact Detection



(c) AF8 Artifact Detection



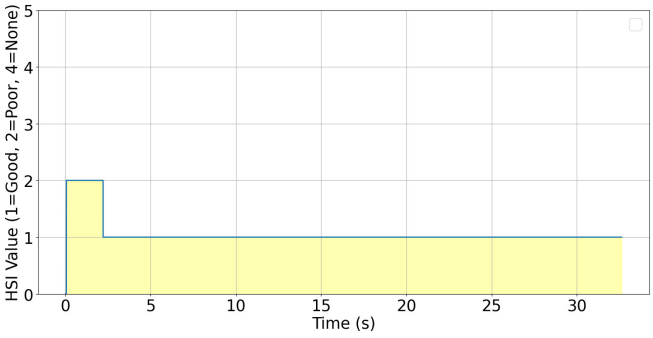
(d) TP10 Artifact Detection

Fig. 7: Artifact detection across electrodes

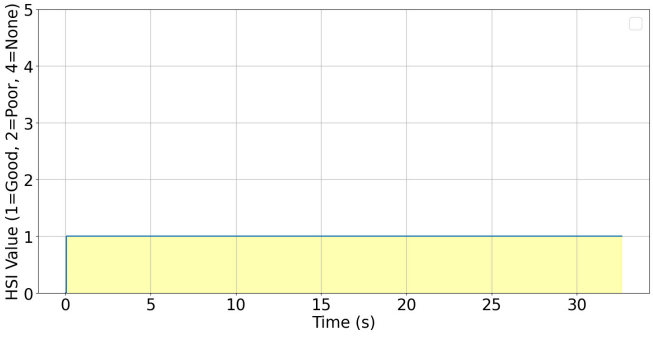
Muse for each electrode, and here we show the average values of relative frequency ranges by electrode in Figure 2.

D. Raw fNIRS & PPG Data

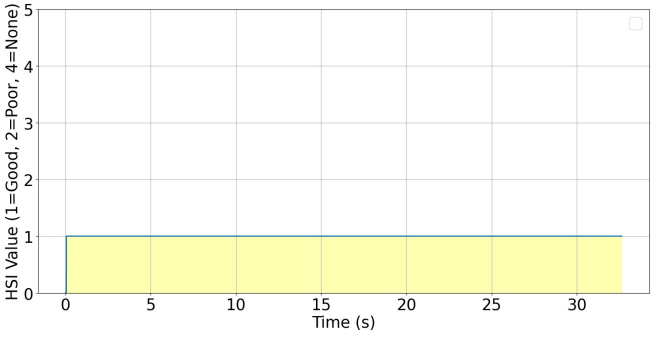
Aside from EEG, some Muse headsets are capable of recording additional biometric data in the form of PPG/fNIRS. Different models of Muse record this data using different methods, with the latest and most advanced being the 16-channel Optics of the Muse S Athena. Here, we graph the 4 of the raw optics data taken from the Muse S Athena in our trial recording. These channels represent the 730nm light measurements, which is a wavelength of infrared light that can be used in fNIRS to analyze hemoglobin in the human brain [10]. These graphs are visible in Figure 10.



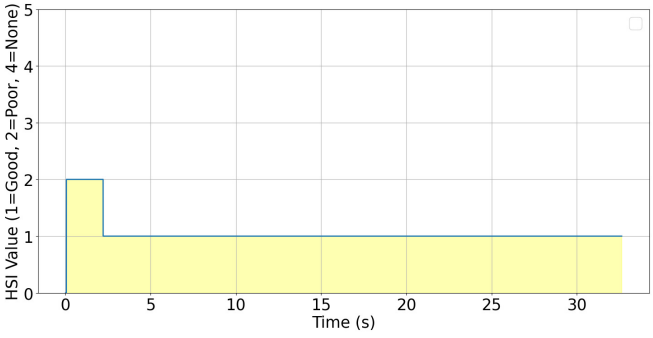
(a) HSI values at TP9



(b) HSI values at AF7

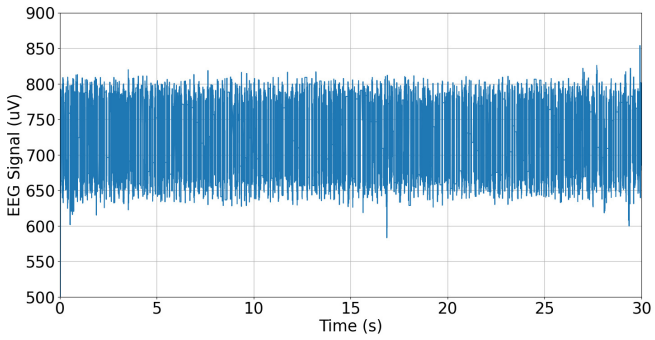


(c) HSI values at AF8

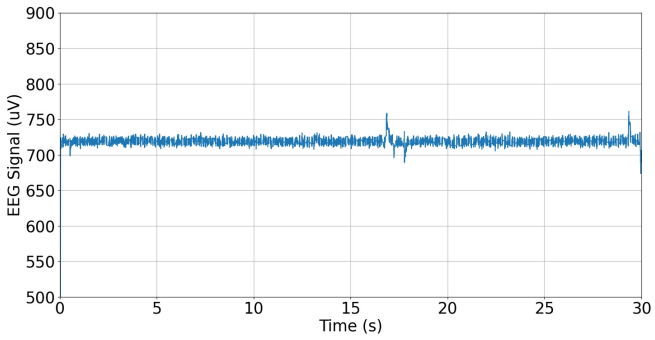


(d) HSI values at TP10

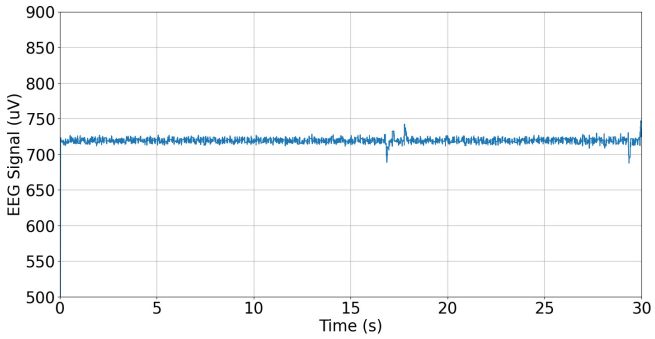
Fig. 8: Headband connection quality across electrodes



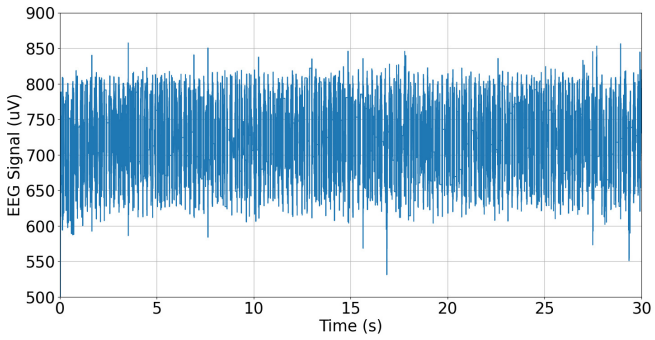
(a) Raw EEG at TP9



(b) Raw EEG at AF7

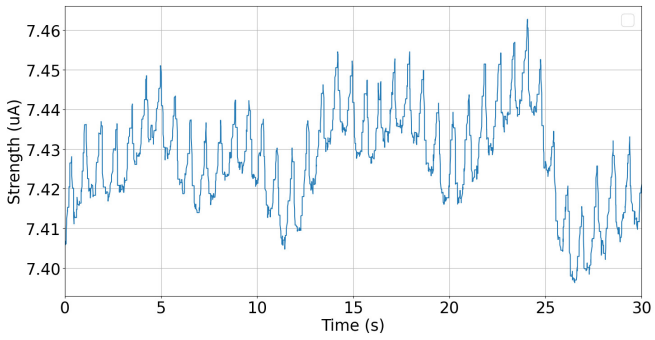


(c) Raw EEG at AF8

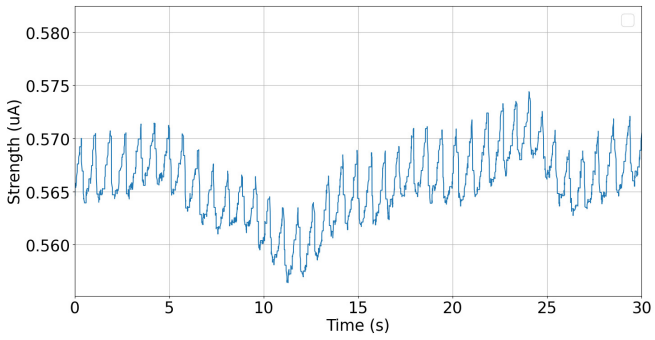


(d) Raw EEG at TP10

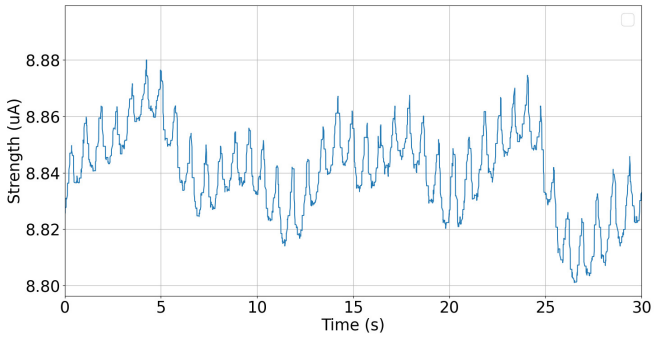
Fig. 9: Raw EEG signals across electrodes



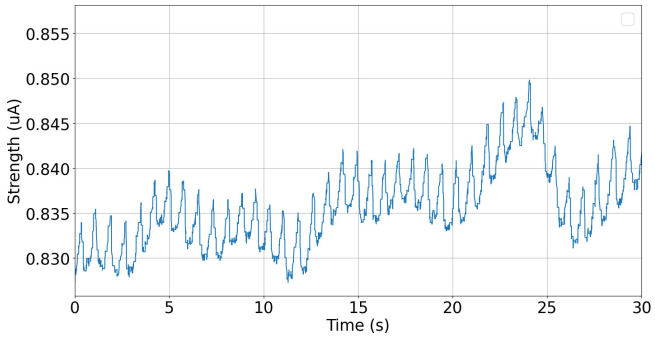
(a) 730 nm Left Inner Channel



(b) 730 nm Left Outer Channel



(c) 730 nm Right Inner Channel



(d) 730 nm Right Outer Channel

Fig. 10: 730 nm Channel Responses Across Inner and Outer Regions

V. Further Research

Advancement in EEG & fNIRS technology has brought us to a point where research can be conducted using low-cost, portable technology like the Muse headbands. The goal of this project, much like previous MuseCroc projects developed at MannLab, is to facilitate and encourage research in diverse settings that may be unsuitable or impractical for traditional EEG or fNIRS systems, in addition to creating easily distributable systems to allow for higher quantities of research done by more people.

VI. Conclusion

This study has demonstrated the feasibility of using the Muse SDK and products developed using it in the collection of data suitable for research into human biometric data using EEG, fNIRS, and other data available from the Muse. The system makes two main contributions: (1) a reusable wrapper around the Muse SDK, reducing access costs for researchers and developers wanting to create their own mobile biosignal applications, and (2) a simple Android app providing real-time visualization and data export into research-friendly formats. The development of similar technology using core components of the program can encourage the proliferation of brain-computer-interface based research and experimentation using easily accessible hardware.

VII. Acknowledgment

Biggest thanks to Prof. Steve Mann for providing me with the opportunity & resources necessary to complete this project. It's been an absolute blast to work with you at MannLab and I'm excited for what comes in the future. Also big thanks to Nishant Kumar for helping me write this paper, Homey Poon for helping me level up my Android programming skills, and Rocklen Jeong for getting me started with graphing EEG data.

- Mitchell Seitz

VIII. Ownership & Collaboration

The Muse is a commercially available intellectual property of Interaxon Inc., and this paper does not imply any ownership or inventorship of the Muse SDK or Muse devices by any other than Interaxon Inc.

"MuseCroc" and/or "MuseCrock" is a term coined at MannLab, Toronto, Canada, and refers to a device that serves the purpose of obtaining data broadcast by a Muse device. "MuseCroc" or "MuseCrock" are thus terms considered trademarked by MannLab, and any individual who would like to use it to represent their work or products must receive permission from Steve Mann of MannLab to do so.

The application MuseCroc Mobile was developed by Mitchell Seitz, with the assistance & support of those listed as co-authors of this paper. Source code contains the Muse SDK, which is property of Interaxon Inc.

If you would like to collaborate, please email me at mseitz@mseitz.dev. Individuals who would like to collaborate or develop using the source code of this application must be authorized to use the Muse SDK by Interaxon Inc.

References

[1] Paolo M. Rossini, Jonathan Cole, Walter Paulus, Ulf Ziemann, and Robert Chen. 1924–2024: First centennial of eeg. *Clinical Neurophysiology*, 170:132–135, 2025.

[2] Nisreen Said Amer and Samir Brahim Belhaouari. Eeg signal processing for medical diagnosis, healthcare, and monitoring: A comprehensive review. *IEEE Access*, 11:143116–143142, 2023.

[3] Cédric Cannard, Arnaud Delorme, and Helané Wahbeh. Identifying hrv and eeg correlates of well-being using ultra-short, portable, and low-cost measurements. *bioRxiv*, 2024.

[4] Steve Mann, Nishant Kumar, Joao Pedro Bicalho, Malek Sibai, and Calum Leaver-Preyra. Adaptive chirplet transform-based sleep state detection. In *2025 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–6, 2025.

[5] Olave E. Krigolson, Chad C. Williams, Angela Norton, Cameron D. Hassall, and Francisco L. Colino. Choosing muse: Validation of a low-cost, portable eeg system for erp research. *Frontiers in Neuroscience*, Volume 11 - 2017, 2017.

[6] Boglarka Vekety, Alexander Logemann, and Zsofia K. Takacs. Mindfulness practice with a brain-sensing device improved cognitive functioning of elementary school children: An exploratory pilot study. *Brain Sciences*, 12(1), 2022.

[7] Georgios Michalareas, Ismat M.A. Rudwan, Claudia Lehr, Paolo Gessini, Alessandro Tavano, and Matthias Grabenhorst. A scalable and robust system for audience eeg recordings. *bioRxiv*, 2022.

[8] Olave E. Krigolson, Mathew R. Hammerstrom, Wande Abimbola, Robert Trska, Bruce W. Wright, Kent G. Hecker, and Gordon Binsted. Using muse: Rapid mobile assessment of brain performance. *Frontiers in Neuroscience*, Volume 15 - 2021, 2021.

[9] Aydin Hosseingholizadeh, Nishant Kumar, and Steve Mann. Portable eeg-based data acquisition and multi-sensor integration using the muse s headband. In *2025 International Conference on Control, Automation and Diagnosis (ICCAD)*, pages 1–6, 2025.

[10] Wei-Liang Chen, Julie Wagner, Nicholas Heugel, Jeffrey Sugar, Yu-Wen Lee, Lisa Conant, Marsha Malloy, Joseph Heffernan, Brendan Quirk, Anthony Zinos, Scott A. Beardsley, Robert Prost, and Harry T. Whelan. Functional near-infrared spectroscopy and its clinical application in the field of neuroscience: Advances and future directions. *Frontiers in Neuroscience*, Volume 14 - 2020, 2020.

[11] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological Measurement*, 28(3):R1, feb 2007.

[12] Steve Mann, Diego Defaz, Tamer Abdulazim, Derek Lam, Mike Alford, Jeremy Stairs, Cayden Pierce, and Christina Mann. Encephalogamestm (brain/mind games): Inclusive health and wellbeing for people of all abilities. In *2019 IEEE Games, Entertainment, Media Conference (GEM)*, pages 1–10, 2019.

Neuromatic: A Multimodal Interface for Mobility Aid Using EEG and Gesture Recognition

Harrison Ossias, Eric Fan, Jessy Ma
University of Toronto
ECE516 - Intelligent Image Processing
{hossias, efan, jma}@mail.utoronto.ca

***Abstract*—Neuromatic is a wearable, multimodal interface that leverages brainwave data and computer vision to provide intuitive, reliable actuation of smart devices. By fusing EEG signals from the Muse S headband with real-time gesture recognition via a wearable Pi Camera module, the system provides a closed-loop control experience for users with mobility impairments. This paper presents the hardware design, communication protocols, and edge inference architecture used to implement Neuromatic.**

I. Introduction

Innovations in industry for technology that automatically interface with the population brings about the question of user-friendliness and surveillance of non-user-controlled technology dominating our lives. Wearable technology such as smart glasses, the Apple Vision Pro and smartwatches have given us a solution to bridge this divide between technology and society. In parallel, devices capable of reading brainwaves, like the Muse EEG headset, have opened up new possibilities in IoT by allowing cognitive input to influence smart systems. By combining brainwave data with computer vision techniques and wearable tech, users can experience an Extended Reality (XR) environment—one where thought and visual input collaborate to provide intuitive, conscious control over surroundings. This project explores the development of such a system, named Neuromatic.

II. Background

One of the major challenges faced by individuals with mobility limitations or impairments from disabilities is accessing appliances and elements around their home. Everyday tasks such as opening doors, cabinets, interacting with high places, knobs and switches can become significant obstacles without assistance. Even smart appliances that offer accessibility services often do so by offering touch control or voice activation solutions, which may not be suitable for all users.

Neuromatic functions with input from hand gesture data collected from a wearable camera and the Muse EEG headset in order to provide real-time decisions on actuation of appliances around the home. By combining gesture recognition with cognitive confirmation, Neuromatic ensures that only intentional actions trigger system responses. This dual-layer approach enhances both accessibility and safety, empowering users to interact with their surroundings in a seamless and dignified way.

III. Hardware

The system integrates EEG, visual sensing, and local actuation into a single wearable pipeline designed for real-time, closed-loop interaction. The hardware is modular and optimized for low-latency, event-triggered control across varied compute platforms.

A. A. EEG Interface

It reads raw brainwave activity on four channels using the Muse S EEG headband. It communicates via Bluetooth Low Energy (BLE) to a local edge server on the actuator.

B. B. Visual Sensing and QR Localization

A Raspberry Pi Zero 2W with a Pi Camera v1.3 attached is powered by a 3.7 V LiPo battery and housed in a PETG 3D-printed case clipped to the front of the Muse headset. This module is activated on BLE proximity to a known actuator and begins to look for QR codes, which are target device addresses or MQTT topics. When a code is detected, the Pi streams video to the provided server for gesture recognition and multimodal fusion.

C. C. Multimodal Integration and Actuation Logic

The actuator node merges EEG and gesture input through a digital "AND" gate approach—actuation will only take place if both modalities agree on intent. Server-side gesture recognition using MediaPipe Hands detects 21-point hand landmark constellations and infer actions (pointing, gripping, etc.). EEG-based intent confirmation prevents accidental activations from casual gestures or noise.

D. E. Enclosure and Power Design

The enclosure design incorporates snap-fit mounting for the internal components, a front aperture for the camera, and ventilation slots for passive cooling. PETG was selected due to its mechanical strength and resistance to UV. The case includes a dedicated slot for a universal snap-fit clip that uses a cantilever arm mechanism for secure attachment to head-mounted equipment. This design enhances stability during movement while allowing quick removal and replacement.

Fig. 1. Front view of the PETG enclosure showing Pi Camera mount and external fasteners.

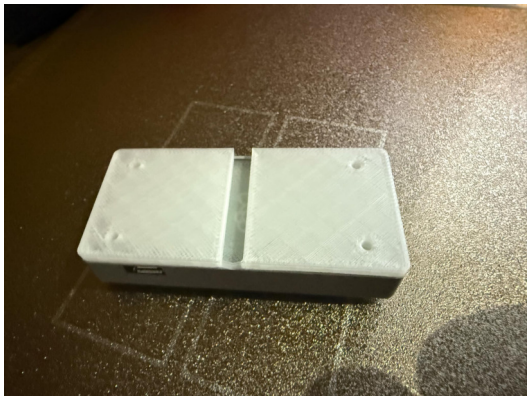


Fig. 2. Rear view of the enclosure with removable snap-fit clip slot and USB access port.

E. F. Edge Performance and Platform Variability

System performance varies by processing device. Tests conducted on a MacBook Pro (M4 28GB model) showed considerable variation in both average response time and stability of the signal. Figure 2 graphs confidence values for EEG and gesture detection across 50 trials, as well as response time per trial.

F. G. Proposed Lightweight Edge Server Library

A modular edge server library is proposed to support future scalability. The following features will be supported:

- BLE session detection
- EEGNet inference engine possibly

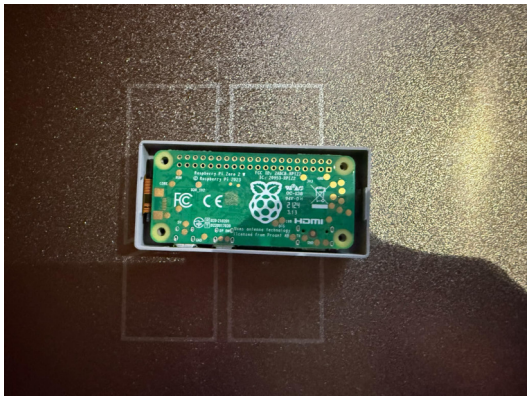


Fig. 3. Internal housing showing the Raspberry Pi Zero 2W mounted inside the enclosure.

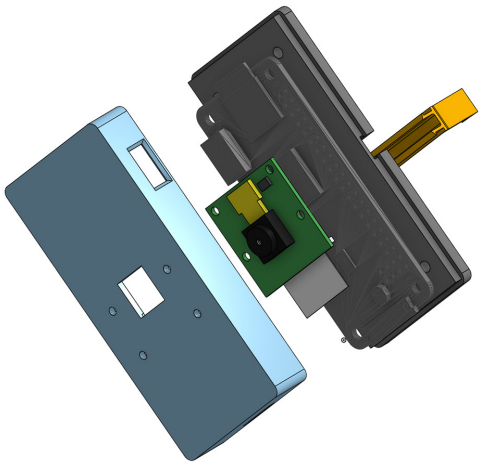


Fig. 4. Exploded and assembled views of the custom PETG enclosure, showing the internal layout of the Pi Zero 2W, Pi Camera, and LiPo battery.

- Upgrade MUSE-LSL to work with the MUSE-S Athena to combine FNIRS
- MQTT with TLS support integration

IV. Communications

A. I. MQTT Protocol Integration

To enable reliable communication between the gesture recognition system and the motor, we utilized the MQTT (Message Queuing Telemetry Transport) protocol.

MQTT is a lightweight publish-subscribe messaging protocol ideal for low-bandwidth and low-power devices, making it a suitable choice for this real-time embedded system. Both OpenCV-based gesture detection and EEG blink classification from the Muse headband act as publisher nodes. When a specific gesture is recognized and a blink event is detected in the EEG signal, a message is published to a topic handled by a Mosquitto MQTT broker, which is either hosted locally or accessible via a networked laptop.

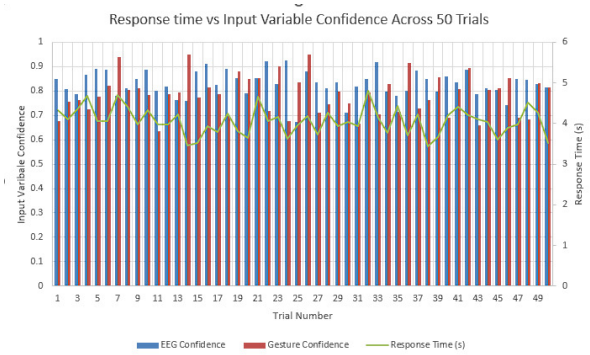


Fig. 5. Plots of EEG and gesture input confidence vs. response time over 50 trials on a MacBook Pro M4 (28GB). Confidence levels are high and consistent, with response times being low (¡200ms) consistently, which reflects the benefits of edge processing with high performance.

B. II. Motor Control by ESP32

The ESP32, acting as an MQTT subscriber, listens for specific messages from the Mosquitto broker. When a command that matches the predefined ”activation condition” (i.e., a valid gesture and blink combination) is received, the ESP32 triggers the motor. This architecture creates a decoupled system where the sensing and actuation modules communicate asynchronously, improving responsiveness and system modularity. Additionally, the MQTT protocol provides Quality of Service (QoS) features that ensure message delivery integrity, which is essential for safety-critical actuation events such as motor control.

V. Software Architecture Implementation

The core vision of NEUROMATIC is to create a seamless interaction flow where users can look at objects they wish to interact with, think about desired actions or focus on those objects, use hand gestures to execute control commands, and receive feedback through both visual confirmation and device actuation. This creates a natural interaction chain that mirrors human cognitive processes - attention, intention, and physical expression - while removing the barriers of traditional input devices.

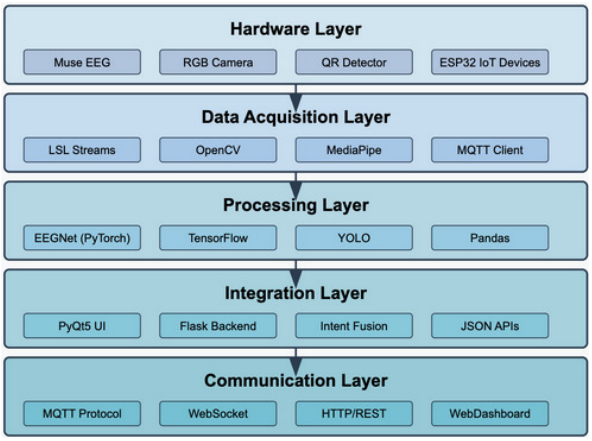
A. 2.1 System Overview

The NEUROMATIC software architecture follows a modular, layered approach with clear separation between data acquisition, processing, integration, and communication components:

layer in the architecture serves a specific purpose: hardware connecting to physical devices, data acquisition interfacing with sensors, processing transforming raw data into meaningful information, integration fusing data sources into commands, and communication transmitting these commands to output devices.

B. 2.2 Input Processing Modules

The system handles three distinct input modalities:



1) 2.2.1 EEG Signal Processing: The EEG processing module handles brainwave data from Muse headsets. It begins with filtering and preprocessing of raw EEG signals to remove noise and artifacts. The cleaned data then undergoes feature extraction using a temporal convolutional neural network EEGNet, which has been specifically designed to identify patterns in brain activity with minimal training data. This PyTorch-based model classifies the brain states into discrete categories that can be used for intent recognition.

2) 2.2.2 Gesture Recognition System: The gesture recognition system uses a standard webcam or pi camera and processes frames through the OpenCV and MediaPipe libraries to extract hand landmarks. These landmarks are then fed into a TensorFlow model that has been trained to recognize specific gestures such as pointing, grabbing, or waving.

3) 2.2.3 Object Recognition and QR Detection: The object recognition module currently focuses on QR code detection as a method for identifying objects in the environment. The system uses computer vision algorithms to scan the environment and detect QR codes, and decode their content. This provides a way to uniquely identify smart devices. In future iterations, this will expand to general object detection using YOLO or similar algorithms to recognize everyday objects without requiring special markers.

C. 2.3 Integration and Decision Making

The heart of the NEUROMATIC system is the integration layer, which combines signals from all three input sources to determine user intent. This fusion algorithm correlates brain activity with visual focus and hand gestures to create an understanding of what the user wants to accomplish. For example, brain activity indicating focus combined with looking at a lamp and making a specific hand gesture might be interpreted as the intent to turn the lamp on or off.

The integration is handled by a Flask-based backend that processes incoming data streams and applies a set of configurable rules to determine when and how to activate connected devices. This rule-based system allows for personalization based on individual preferences and capabilities.

D. 2.4 Dashboard and Control Interface

A web-based dashboard provides real-time visualization of all system components and allows users to define custom triggers and actions. Through this interface, users can:

- Monitor their EEG signals in real-time with visual feedback
- See recognized gestures as they perform them
- Define custom combinations of inputs that trigger specific actions
- Configure new devices and control schemes

The dashboard is built with Flask and WebSocket to ensure responsive feedback.

E. 2.5 Output and Device Control

The final stage is the communication layer, which transmits commands to smart devices using the MQTT protocol. The system is currently designed to control ESP32-based smart home components but can be extended to any device that supports MQTT or similar protocols.

VI. Conclusion

Neuromatic provides an interface between society and technology via human input and realizes XR (Extended Reality) while providing a solution for individuals with mobility impairments. Neuromatic differs from traditional smart systems which often automate tasks like opening doors or controlling appliances, but which frequently lack user-centered design—especially for those unable to use voice or touch interfaces. By confirming deliberate intent through EEG signals, Neuromatic enables users to intuitively and safely control their environment via a blend of physical gestures and cognitive input. Wearable technology such as Neuromatic responds to the global culture of surveillance by a system of sous-veillance, empowers users with greater independence, globalizes the use of XR and provides a higher quality of life for many.

Future Work

Looking ahead, several key developments are planned to enhance the capabilities and accessibility of the Neuromatic system.

First, implementing a BLE (Bluetooth Low Energy) range-to-actuator finder will enable the system to dynamically identify and connect to nearby smart devices, allowing for more context-aware interactions within the home. BLE is already a commonly used method for communication in wearable technology and IoT systems, due to its low power consumption and ability to maintain stable connections for short distances (10-30 meters indoors). As such, we would like to scale the design of Neuromatic to accommodate BLE.

Additionally, to encourage community-driven innovation and wider traction/popularity, the Neuromatic software development kit (SDK) and hardware housing designs will be open-sourced, providing developers and makers with the tools to customize and expand the system.

Finally, the reliability of the EEG data is dependent on large-scale data collection. This will be prioritized to capture a diverse range of EEG signals and gesture patterns, which will help improve the inclusivity of the system. With more data and further tuning of the machine learning models for classification, the accuracy of gesture will be refined, ensuring robust performance across various users and environments.

References

[1] Google. "MediaPipe Hands." [Online]. Available: https://mediapipe.readthedocs.io/en/latest/solutions/hands.html

[2] Google. "3D hand pose estimation using MediaPipe and TensorFlow.js." Nov. 2021. [Online]. Available: https://blog.tensorflow.org/2021/11/3D-handpose.html

[3] Gatoninja236. "Scan QR Codes in Real-Time with Raspberry Pi." Hackster.io. [Online]. Available: https://www.hackster.io/gatoninja236/scan-qr-codes-in-real-time-with-raspberry-pi-a5268b

[4] Google. "Handtrack.js: Tracking hand interactions made easy." Nov. 2019. [Online]. Available: https://blog.tensorflow.org/2019/11/handtrackjs-tracking-hand-interactions.html

Spinal Drift: Research on Dynamic Core Engagement Through Interactive Gameplay

Jonathon Wang, Chahit Uppal, Nicholas Wood

***Abstract*— Spinal Drift is an interactive physiotherapy game designed to promote core strength and spinal health through engaging, sensor-driven gameplay. By transforming a traditional plank exercise into a competitive racing experience, the game leverages the inertial measurement unit (IMU) sensors in smartphones to track users' core movements in real time. Players control an on-screen vehicle by shifting their body while in a planking position on a balance board, seamlessly integrating exercise with immersive gameplay. The system features a React Native mobile app for motion tracking, WebSocket communication for low-latency data transfer, and a browser-based 3D game environment powered by Three.js. Preliminary user testing showed high engagement, rapid adaptation to controls, and positive feedback, with many participants unaware they were exercising due to the fun, competitive design. Future developments include multiplayer modes, adaptive difficulty, and clinical validation in partnership with healthcare professionals. Spinal Drift demonstrates the potential of gamified rehabilitation tools to transform physiotherapy into an accessible and enjoyable experience.**

I. Introduction & Background

Physiotherapy has come a long way in recent years, especially with the rise of interactive and gamified tools designed to make exercising more engaging. Spinal Drift was created right at this crossroads of health tech and gaming, offering a fresh take on spinal and core health.

The spine plays a vital role in our posture, movement, and daily activities. When core muscles are weak, it can lead to bad posture, back pain, or even injuries. While exercises like planks are great for building core strength, they are often not fun and exciting. That's where Spinal Drift steps in. Using the motion sensors (gyroscope) built into smartphones, it turns a simple core workout into a fun, interactive racing game. Players get into a plank position on a balance board and steer the game with subtle shifts in their core.

By merging motion-tracking tech with physiotherapy, Spinal Drift makes working out feel less like a chore and more of a challenge to overcome through competitive gameplay elements. It encourages players to stay engaged and build their core strength, all while having a little fun along the way.

II. Gameplay

In the Spinal Drift game, the player controls a vehicle facilitated by an inertial measurement unit (IMU) sensor integrated with their phone. The car has three controls: left and right for respective tilting directions and the ability to slow down through forward and backward tilting. Utilizing the balance board to place the phone (see Figure 1), the player will enter a planking position and shift their core based on how they want to steer the vehicle. The core concept is combining physiotherapy exercises such as planking to encourage spinal health through strengthening the core via an engaging interactive racing game.

Fig. 1. Balancing board with IMU phone app used for game controls

A. User Menu Settings

Before beginning the game, the user is prompted with a menu interface for various settings and information that they can choose to read (see Figure 2).

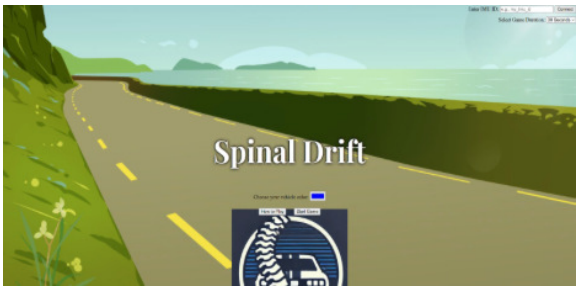


Fig. 2. Spinal Drift user menu interface

The primary setting that the player must have before beginning the game is calibrating their IMU ID. This ID is determined by the IMU app used as the controller of the game, in which both tags must share the same name to establish a connection after clicking on the "Connect" button (see Figure 3).

Fig. 3. Verify matching IMU ID before connecting

Due to Spinal Drift's endless runner gameplay, the game is only over after the time limit has run out. The time limit is determined by the game duration set by the player based on their exercise comfort level (see Figure 4). If no time is selected, the game will run at a default duration of 30 seconds.

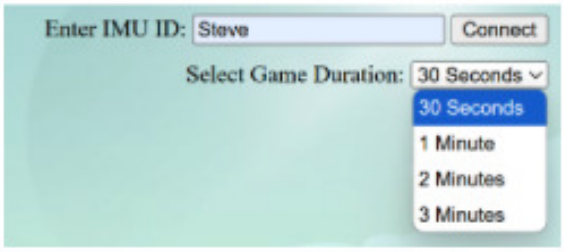


Fig. 4. Drop-down menu for player selection of game duration

For new players unsure of specific game mechanics, a user summary is provided by clicking the "How to Play" button. This allows for new players to learn the game through brief informative instructions (see Figure 5).

Fig. 5. Gameplay information pop-out window for new users to read

The last setting on the menu is the customization feature that allows the player to choose the color of their vehicle based on their preference (see Figure 6). If no color is selected, the game will load with the default blue color setting. When all changes are set by the user, the player can click the "Start Game" button to load Spinal Drift and begin the game.

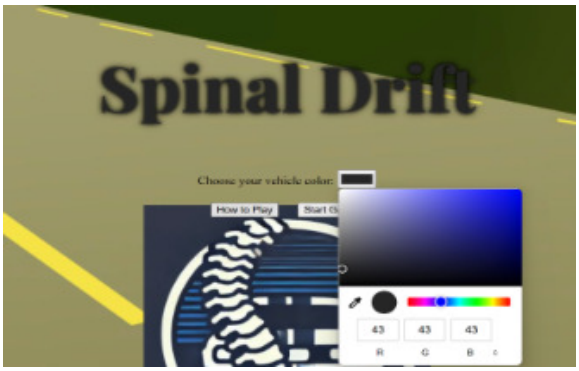


Fig. 6. Customizable user color settings of game vehicle

B. Road Conditions

Three main driving conditions will be cycled during the duration of the game, each offering a unique set of exercise challenges that the player will face, listed as follows: traffic roads, construction zones, and curved roads. The traffic road condition is the most common driving environment that the user will come across, which trains the player's reactive movement during the exercise. During traffic, the car must try to quickly dodge the opposing vehicles parked on the road by dodging either left or right (see Figure 7). The player will get point deductions based on the challenges they fail during this segment. If the vehicle hits any cars when driving, the player will be penalized with a "-2" point deduction (see Figure 8). Additionally, if the player swerves to the left or right and reaches the borders of the road, they will experience a "-1" point deduction (see Figure 8).

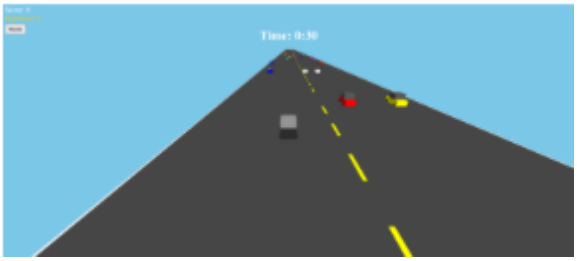


Fig. 7. Traffic Road Environment Condition

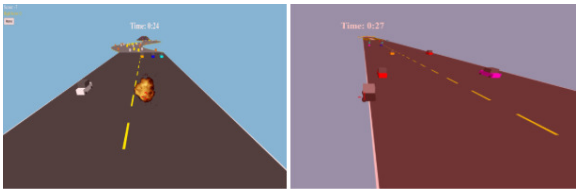


Fig. 8. Traffic Road Environment Condition

To ensure stability and balance during the exercise, the construction zone conditions encourage users to stay aligned with the road. In this segment, the player is awarded for little to no movement, ensuring the car does not veer off the road. When the vehicle is continuously driving on the road without drifting off, the game awards "+2" points as long as the user stays in the center (see Figure 9). However, the zone also penalizes the player depending on how off-center the car is. Four unsafe regions within this zone deducts "-1", "-2", "-3", and "-4" points, respectively (see Figure 10). In addition to the penalty points, the speed of the vehicle also decreases, making it drive slower as it moves further and further off-road.

Fig. 9. Construction zone environmental condition with points awarded for alignment

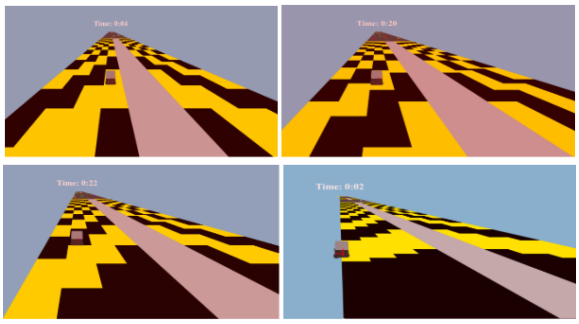


Fig. 10. Penalty for veering off-road based on drift severity from center

The last road condition that the user will experience is a winding road that focuses on gradual, sustained tilts for line tracking. This road environment intends to train the user with steady movement coordination during the exercise with an award system based on how well the player stays in the middle. The gold coin spawns in the middle of the road, which provides "+3" points if collected (see Figure 11). If the car is not aligned with the middle of the track, the player can still obtain silver coins that reward "+2" points and copper coins that give only "+1" points (see Figure 12). To penalize severe failing of coordination, there will be points deducted if the car touches the borders of the curved road that will "-2" points from the player (see Figure 13).

Fig. 11. Curving road condition with points awarded for accurate line tracking

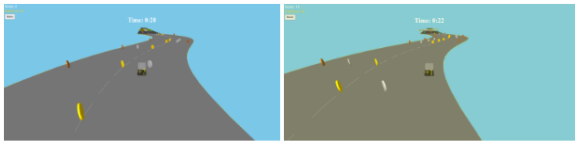


Fig. 12. Less points awarded for inaccurate line tracking with center of winding road

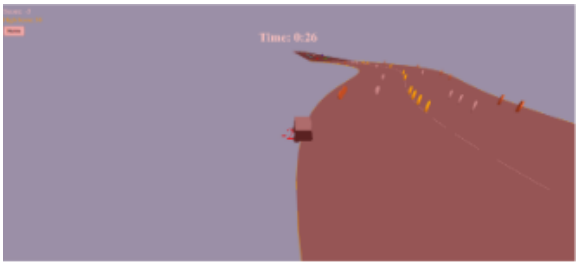


Fig. 13. Border penalty for coordination failure

III. High-Level Design

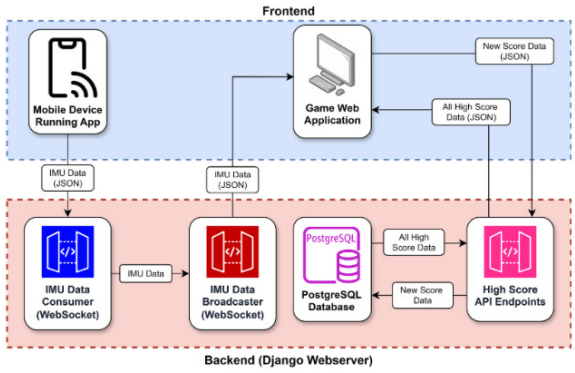


Fig. 14. Border penalty for coordination failure

Spinal Drift is built around three main pillars: hands-on user interaction, core-driven motion control, and gradually increasing difficulty through changing terrain. At the heart of it all are the smartphone's IMU sensors, which pick up on how the phone tilts and moves.

Everything kicks off with a simple, intuitive menu. Here, users can tweak how their vehicle looks and choose how

long they want to play. Once the game starts, players dive into timed challenges that change based on the "road" they're driving on. The IMU tracks body movements in real time, letting players control the vehicle by shifting their weight, like steering with your core.

The game features three environments that switch on the fly: busy traffic streets, tricky construction areas, and curvy mountain roads. Each one is designed to work different parts of your core and test things like posture, balance, and reaction speed. There are rewards for good control and penalties for poor form, all built into the gameplay to keep players focused and motivated.

A lot of what makes Spinal Drift work comes down to fast, accurate data processing. The motion data is sent through a local connection and interpreted by the game's physics engine, which reacts instantly to the player's movements. That way, what you do physically matches exactly what you see on screen. This helps build a stronger connection between body control and performance.

IV. Technical Research

The core of this system is a cross-platform mobile app built with React Native and the Expo framework. This combo was chosen because it's quick to develop and works smoothly across both iOS and Android devices. The app taps into the phone's gyroscope and accelerometer to continuously track movement, specifically along the pitch, roll, and yaw axes.

That motion data gets bundled into neatly structured JSON packets at around 60 times per second, ensuring a steady stream of information. It's then sent through WebSockets, a communication method chosen for its low latency and ability to keep a constant connection open between the phone and the server. This setup helps ensure that every slight movement the user makes is captured and reflected in the game almost instantly.

On the other end, the server listens for incoming WebSocket messages and interprets them as in-game controls. These inputs are fed into a 3D simulation powered by Three.js that runs in the browser and handles all the visual rendering. Three.js allows Spinal Drift to create detailed environments, simulate realistic vehicle dynamics, and keep the game responsive on a variety of devices.

The steps in our technical design are as follows:

1) **Sensor Capture (React Native App)**
   - Uses Expo's DeviceMotion and Gyroscope APIs
   - Collects orientation data at $\sim$60 frames per second
   - Smooths the data with a moving average filter to reduce noise
2) **Data Transmission (WebSocket)**
   - Sends the motion data through a persistent WebSocket connection
   - Includes keep-alive signals to keep the connection stable
3) **Game Engine (Three.js)**
   - Maps the motion data to control how the vehicle moves
   - Adjusts things like speed, turning, and braking based on tilt
   - Renders road types like traffic zones, construction sites, and winding paths
   - Handles physics, collisions, and the points system in real time

The system was also designed to be flexible. Developers can tweak things like tilt sensitivity and response thresholds, which makes it easier to add future features like adaptive difficulty or personalized workout modes. All of this adds up to a smooth, immersive gameplay experience with plenty of room to grow.

V. Future Improvements

To further refine the user experience in making it both more interactive and therapeutic, the team has begun looking into advancements for Spinal Drift. These include optimizing the sensor technology of the IMU, which will allow for more accurate monitoring of posture and core engagement. One method to do this is through including adaptive calibration systems that will allow the user to tune the sensitivity of the tilt controls that best suit their exercise levels and offer personalized difficulty settings. Furthermore, in terms of gameplay, the team has discussed the implementation of multiplayer functionality that will increase engagement through friendly competition. These planned changes will add in a global and local leaderboard that can be used to boost engagement in the game by achieving the highest score. Another multiplayer option includes allowing two players to drive on the same map and interact with one another through power-up items that can decrease the opposing player's score or be used to increase their score.

Once the game has been fully optimized based on the changes, the team plans to seek professional research with healthcare professionals who can help integrate the game clinically through systems and physiotherapy protocols that ensure the gameplay contributes effectively to spinal health. Through incorporating gameplay changes with the addition of professional research, Spinal Drift can be finalized to be a robust, interactive platform that facilitates long-term rehabilitation along with core strengthening.

VI. Conclusion & Final Results

The final showcase of Spinal Drift at the symposium was a major success, both technically and in terms of user engagement. One of the clearest signs that the project struck a chord was how many people came back for second and even third rounds.

We noticed a consistent pattern: players improved with each attempt. This supported our belief that users would quickly adapt to the IMU-based controls and become more confident in steering the game with their core movements. We had set a benchmark score of 50 as a sign of solid performance, and most players were hitting or exceeding that after just a couple of rounds. The learning curve was smooth, and the game's mechanics felt intuitive even for first-time users.

Player feedback was overwhelmingly positive. Many people were surprised to realize they had been planking or engaging their core the entire time, something that felt more like play than exercise. These positive reactions verified the project goals of what we set out to achieve: turning physiotherapy into something enjoyable and immersive.

On the technical side, the system held up impressively well. The WebSocket data connection stayed fast and stable, even with multiple users playing in quick succession. The React Native app ran smoothly across different phones, and the Three.js graphics engine kept visuals sharp and responsive. Even with sustained play, latency was low, and the controls remained accurate. The setup proved it could handle a fast-paced, wireless rehab-style game with real-time sensor input and 3D visuals.

In the end, Spinal Drift delivered exactly what we hoped for: a meaningful blend of gameplay and therapeutic motion. The strong user response, consistent engagement, and solid technical performance all point to its real-world potential as a scalable and effective tool in interactive physiotherapy.

ACKNOWLEDGEMENTS

We would like to extend our sincere thanks to Professor Steve Mann for providing the balance board equipment and for inspiring the original idea behind an immersive final project that merges interactivity with positive health outcomes. His guidance throughout the course played a key role in helping us navigate the space where technology, user experience, and physical well-being come together. We are especially thankful to our TA, Nishant Kumar, for his continuous support, valuable feedback, and technical insights throughout the project. His assistance was instrumental in troubleshooting development challenges and refining the system. We are also grateful to the Department of Electrical and Computer Engineering at the University of Toronto for offering the ECE516 course. The technical support, mentorship, and creative freedom we received were invaluable in bringing Spinal Drift from a concept to a fully realized, engaging, and impactful project. The course's collaborative and forward-thinking atmosphere made all the difference in our development journey.

Adaptive Chirplet Transform and Deep Learning Algorithms for EEG based Sleep Stage Detection

João Pedro Bicalho Andrade, Steve Mann

I. Abstract

Electroencephalography (EEG) is a powerful tool for analyzing brain activity, particularly in the context of sleep stage detection. However, traditional signal processing methods like the Fourier and Wavelet Transforms often struggle to capture the non-stationary and time-varying nature of EEG signals. This thesis explores the Adaptive Chirplet Transform (ACT) as a novel feature extraction technique for EEG-based sleep stage classification using deep learning. The ACT offers a dynamic time-frequency representation by matching signal segments to Gaussian chirplets. A GPU-accelerated implementation of the ACT was developed and applied to the Bitbrain Open Access Sleep Dataset, processing over 10,000 epochs from three nights of sleep. Feature vectors extracted by the ACT were reshaped into 5×5 matrices and used as input to a hybrid Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU) model. The model achieved a training accuracy of 91.5% and a test accuracy of 57.9%, demonstrating the ACT's potential to encode relevant features with a 153.6:1 compression ratio. The findings suggest that, with further optimization and scaling, the ACT could become the next state of the art signal processing and feature extraction method for EEG.

II. Introduction

Building on the diverse methodologies for EEG analysis, this thesis specifically focuses on the application of the Adaptive Chirplet Transform combined with deep learning to detect sleep stage. Sleep stage detection is a critical area of research due to its significance in diagnosing sleep disorders, understanding brain function during rest, and developing interventions to improve sleep quality. Deep learning offers a transformative approach by automating the extraction of meaningful features from complex, high-dimensional EEG data. Unlike traditional manual scoring methods or simpler classification techniques, deep learning models do well at capturing sophisticated temporal and spatial patterns within EEG signals. By using data from publicly available datasets, this study aligns with broader scientific practices for benchmarking and comparability while avoiding the logistical and ethical complexities of collecting new EEG data.

Techniques such as the Fourier Transform and Wavelet Transform have traditionally dominated time-frequency analysis in EEG research. The Fourier Transform decomposes signals into their constituent frequencies, providing valuable insights into rhythmic brain activity. Wavelet Transforms extend this by combining frequency and temporal information, enabling the analysis of dynamic changes over time. However, these methods face limitations when capturing complex, non-stationary signal features, such as rapid frequency modulations or localized transient events. This paper adopts the Adaptive Chirplet Transform (ACT) as a promising alternative. [1] [2], [3] Unlike the fixed basis functions of Fourier and Wavelet Transforms, the ACT dynamically adapts to the signal's local characteristics, offering a more flexible and precise representation of time-varying EEG patterns. These qualities make the ACT particularly suitable for sleep stage detection, where subtle physiological changes manifest as transient shifts in frequency content.

Despite its potential, the application of the ACT to EEG data remains under-explored, with only a few studies demonstrating its capabilities. [2], [4]–[12] This gap presents an opportunity to advance EEG research by integrating the ACT with state-of-the-art deep learning architectures. The ACT's ability to provide a detailed, dynamic representation of EEG signals enhances the input to deep learning models, potentially improving their classification accuracy and robustness. Sleep stage transitions, which involve nuanced changes in frequency and amplitude, are especially well-suited for analysis using the ACT.

The primary objective of this thesis is to address this research gap by employing the ACT as the feature extraction algorithm for the EEG data and feed it to a Deep Learning architecture for the application of sleep stage detection.

III. Background and Literature Review

A. Introduction to EEG Analysis

Electroencephalography (EEG) is a widely utilized technique that measures the electrical activity generated by the brain, allowing for brain wave visualization. Specifically, it detects minute differences in electric potential at the scalp, which result from the collective

activity of post-synaptic potentials produced by neurons within the cortical layers.

EEG data is used extensively to analyze brain activity, but EEG signals are highly complex due to their non-stationary nature, low signal to noise ratio, and the presence of artifacts from non-neural sources such as muscular activation. Even after processing, the data is still extremely intricate and convoluted, making it difficult to interpret. To extract meaningful information from EEG signals, the data needs to go through rigorous processing and analysis.

B. Traditional Methods in EEG Analysis

The analysis of EEG signals spans diverse methodologies, each suited to different applications and offering unique insights into brain activity.

1) Event-Related Potentials (ERPs): Event-Related Potentials (ERPs) represent one of the most established approaches, involving the identification and analysis of brain responses that are time-locked to specific events, such as sensory stimuli or cognitive tasks. Examples include the P300, a positive deflection occurring roughly 300 ms after an expected stimulus, often used in BCIs and studies on attention and decision-making, and Steady-State Visual Evoked Potentials (SSVEPs), which leverage repetitive visual stimulation for applications like assistive technology and gaming. ERPs are widely used in cognitive neuroscience due to their ability to isolate specific neural processes. However, they require significant manual preprocessing and are often limited by their reliance on predefined stimuli and rigid temporal windows, making them less adaptable to complex, real-world scenarios.

2) Spectral Analysis: Spectral analysis, another cornerstone of EEG research, focuses on the frequency content of brain signals to reveal insights into oscillatory activity across delta, theta, alpha, beta, and gamma bands. Each frequency band correlates with distinct cognitive and physiological states, such as delta waves linked to deep sleep or alpha waves associated with relaxation and attention. By employing time-frequency techniques like Wavelet Transforms or Chirplet Transforms, spectral analysis bridges the gap between traditional ERPs and modern deep learning by enabling the study of dynamic changes in oscillatory activity. It finds extensive use in both clinical and research settings, such as monitoring epilepsy, understanding attention dynamics, and tracking meditation progress. [13]

3) EEG Data Pre-processing, Filtering and Artifact Removal: Noise and artifacts from non-neural sources can be broken down into physiological noise and environmental noise. Physiological sources contributing the most noise come from eyeball movement, which is known as electrocorticogram (EOG), the cardiac signal, known as electrocardiogram (ECG), and muscular contractions, known as electromyography (EMG). Environmental noise can encompass electromagnetic fields in the vicinity caused by any other non biological sources, such as those caused from AC power lines or electronic devices in the room.

Signal processing methods are employed to address these challenges, enhancing the interpretability of EEG data. Signal Processing is a broad field and active area of research, and in the context of EEG signals it encompasses several different tasks, most notably pre-processing, feature extraction and analysis and interpretation of data. Pre-processing includes data filtering for noise reduction and artifact removal, segmenting the data into appropriate epochs and normalizing the data.

Filters such as low-pass, high-pass, band-pass, and notch filters are commonly used to retain relevant brainwave frequencies while suppressing noise from sources like eye movements or power line interference. By focusing on frequency-specific manipulations, data filtering provides a clean, noise-reduced signal, forming the basis for subsequent, more sophisticated signal processing techniques.

Preprocessing EEG data is a critical step for the success of deep learning models. Filtering techniques include bandpass filtering to eliminate frequencies outside the range of interest (e.g., 0.5–50 Hz for most EEG tasks) and notch filtering to remove powerline interference such as 50 or 60 Hz noise. Artifact removal methods are employed to mitigate non-brain artifacts, utilizing techniques such as independent component analysis (ICA) to separate mixed signals and isolate artifacts, regression-based methods to subtract artifacts like eye blinks using reference signals, and consensus filtering, which applies multiple filtering methods to ensure signal integrity. Signal standardization, including normalization or z-scoring, ensures consistency and improves convergence during training. [13]

C. Adaptive Chirplet Transform (ACT)

The ACT was originally developed by S. Mann and S. Haykin in 1991 and since then several papers have proposed modifications or possible improvements to it. [3] Some notable examples include the "Adaptive Linear Chirplet Transform" by Guan et Al. the "Multi-Scynchrosqueezing Chirplet Transform" by Zhu et Al. [14] the "Enhanced Adaptive Linear Chirplet Transform" (EALCT) by Lopez et Al. [7] and lastly, developed by Chui et Al., the Chirplet Transform-based signal separation scheme (CT3S). [15]

Given its flexibility and capability to capture information on signals that vary both in time and frequency, the ACT has also been used across a wide range of applications. From aircraft bearings fault diagnosis, to

P300, VEPs and epileptic seizure detection with EEG, target recognition from RADAR technology, and more. [2], [9], [16]–[18]

After a thorough analysis of several different versions and models of the ACT it was clear to see that the paper which aligned the most with the goals set out by this thesis was the work of Bhargava et Al. Besides using the ACT to analyze P300 signals from EEG data, Bhargava also published an open source Python implementation of the ACT optimized with the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm.

The implementation works by creating an ACT class that is able to perform the Adaptive Chirplet Transform on any given signal. The class initializes by either generating a dictionary of Gaussian chirplet functions or loading one from the cache if one has already been generated. The size of the dictionary is variable, and it depends on some of the parameters passed in by the user when creating an instance of the class.

The equation for a general Gaussian Chirplet is shown below:

$$g(t) = \frac{1}{\sqrt[4]{\pi}\sqrt{\Delta_t}} \exp\left\{ -\frac{1}{2}\left(\frac{t-t_c}{\Delta_t}\right)^2 + j \cdot 2\pi \left[c(t-t_c) + f_c(t-t_c)\right] \right. \tag{1}$$

The Gaussian chirplet above has a center time t_c, duration Δ_t, frequency modulation (chirp) rate c, and center frequency f_c. These four parameters are what is needed to fully characterize a Gaussian chirplet function. The dictionary then is constructed by generating different Gaussian chirplets varying these parameters over a range and step size defined by the user when instantiating the class.

Once the dictionary has been generated, the ACT compares the given signal to the Gaussian Chirplets in the dictionary, and it picks the one which most closely matches the signal, using the BFGS optimization algorithm to do so. This step is called a first order Chirplet Transform, and this step can be repeated as many times as desired, achieving a higher order ACT. More concisely, this method represents a signal as a linear combination of *chirplets*, which are Gaussian-windowed, frequency-modulated basis functions.

For an N order ACT, this implementation returns several lists. First a list of coefficients, where the n^{th} element of the lists represents the coefficient that scales the best match Gaussian Chirplet chosen in the n^{th} iteration of the algorithm. Followed by this it returned a list of size $4 * n$, where the elements are the four parameters necessary to identify each of the chosen Gaussian Chirplets from the dictionary. For the purposes of applying the ACT, these two lists are the most important, but it also provids a residue list and a raw error list, both of which could be used as a measure of how accurately the signal was reconstructed by the chirplet transform at different orders.

D. EEG and Deep Learning

Deep learning has emerged as a powerful tool for EEG analysis because it can automatically extract meaningful patterns and features from raw data without requiring extensive manual processing or domain-specific feature engineering. It has enabled breakthroughs in tasks that were previously difficult or impossibly unpractical to achieve using conventional methods. Furthermore, recent advances in deep learning architectures and computational resources keep accelerating progress in this field.

Deep learning has been widely applied to various EEG classification tasks, which can be broadly categorized into domains such as sleep stage classification, mental state and emotion recognition, disease detection and diagnosis, and brain-computer interfaces (BCIs).

Sleep stage classification involves identifying distinct sleep stages, such as NREM and REM, based on EEG patterns, facilitating research in sleep medicine and sleep disorders.

Mental state and emotion recognition tasks focus on classifying cognitive states (e.g., focused vs. relaxed) and emotions (e.g., happiness, stress), often utilized in human-computer interaction applications.

Disease detection and diagnosis tasks aim to identify neurological disorders like epilepsy, Alzheimer's disease, and Parkinson's disease, with seizure detection from EEG signals being a prominent application.

Lastly, BCIs enable control of external devices using EEG signals, involving tasks such as motor imagery classification and control signal generation.

These applications showcase the versatility and transformative potential of deep learning in EEG analysis.

1) Input to Neural Networks: The choice of input data for neural networks is a critical factor in determining the performance of EEG analysis models. Different research approaches vary in their use of raw EEG signals, preprocessed data, and transformed representations to optimize neural network training.

While some studies utilize raw EEG signals as input relying simply on the feature extraction power of the Neural Network, the vast majority of research emphasizes preprocessing and filtering as essential steps. Techniques such as bandpass filtering, notch filtering to remove powerline interference, and independent component analysis (ICA) for ocular and muscular artifact removal are widely employed. These methods enhance signal quality and ensure cleaner inputs for neural networks.

However, beyond initial preprocessing and filtering, there is little consensus among researchers on the extent of artifact removal or the specific techniques to employ. For instance, some studies apply rigorous artifact removal to eliminate non-neuronal signals entirely, while others retain certain artifacts to preserve additional features of the raw signal.

A recent review of 89 studies on deep learning for EEG classification highlights this lack of consensus and variability in pre-processing strategies. [19] The accompanying visual graph from the review illustrates the proportion of studies adopting different artifact removal methods, noting that the majority of studies either did not specify or did not use removal methods.

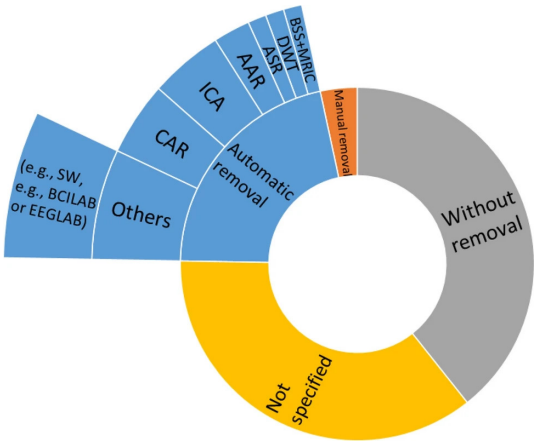


Fig. 1: Artifact removal methods in Deep Learning EEG classification, as summarized in a review of 89 studies.

Transformations of EEG data into alternative domains further diversify input preparation strategies. Frequency-domain techniques, such as the Fourier Transform, decompose the signal into its spectral components, providing insights into frequency-specific brain activity. The Wavelet Transform, offering a time-frequency representation, is frequently utilized to capture transient and non-stationary patterns in EEG signals. These transformed representations can augment neural network models by emphasizing specific aspects of brain activity relevant to the classification task. [20]

Figure 1 is complemented by another systematic review that analyzed 154 studies on deep learning-based EEG analysis. [21] This review explored different input strategies and neural network architectures, shedding light on trends across various methodologies. The insights from these reviews reinforce the importance of aligning input preparation strategies with specific research objectives.

This research will focus on using the Adaptive Chirplet Transform (ACT) as the chosen signal processing method due to its promising results in capturing time-frequency characteristics in other domains. The ACT's ability to provide detailed temporal and spectral representations aligns well with the goals of this study, offering a powerful alternative to traditional signal processing and feature extraction techniques. By applying the ACT, the aim is to enhance the quality of the inputs to the neural networks and explore its potential to improve EEG and Deep Learning based sleep stage classification performance.

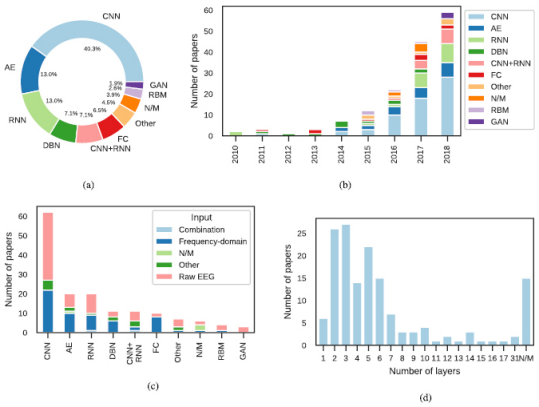


Fig. 2: Deep learning architectures across 154 studies analyzed in "Deep learning-based electroencephalography analysis: a systematic review". 'N/M' stands for 'Not mentioned' and accounts for papers which have not reported the respective deep learning methodology aspect under analysis. (a) Architectures. (b) Distribution of architectures across years. (c) Distribution of input type according to the architecture category. (d) Distribution of number of neural network layers.

2) Deep Learning Architectures: Deep learning architectures play a crucial role in determining the performance and adaptability of models used for sleep stage detection with EEG data. The aforementioned review article of 89 studies on deep learning for EEG classification provides valuable insights into the variety of architectures employed in the field. [19] The findings are summarized in Figure 3, which illustrates the distribution of different deep learning architectures across these studies.

The figure above highlights the diversity of approaches, including CNNs, RNNs, LSTMs, GANs, VAEs, and hybrid models, among others.

The landscape of deep learning applications for EEG-based sleep stage detection is marked by a variety of innovative approaches. Each architecture offers unique strengths and capabilities, reflecting the diverse challenges associated with analyzing EEG data. From capturing intricate spatial patterns to modeling temporal

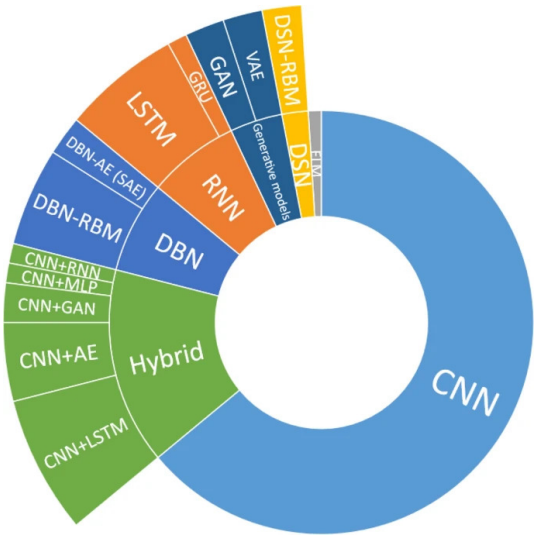


Fig. 3: Deep learning architectures utilized in EEG sleep stage detection.

dynamics, the choice of architecture is often influenced by the specific requirements of the task, the nature of the data, and the desired level of interpretability. By exploring the relative merits and limitations of these architectures, researchers aim to uncover optimal strategies for improving classification performance and advancing the state of the field.

Convolutional Neural Networks (CNNs) are widely used for their ability to automatically extract spatial features from EEG data. [22] They excel in handling structured grid-like data and are particularly effective for time-series and spectrogram-based EEG representations. Many studies employed CNNs in pure form or hybridized with other architectures to enhance performance. Recurrent Neural Networks (RNNs), designed for sequential data, capture temporal dependencies in EEG signals and are particularly suited for tasks requiring the modeling of temporal dynamics, such as sleep stage transitions. Long Short-Term Memory Networks (LSTMs), a specialized type of RNN, address the vanishing gradient problem, enabling them to capture long-term dependencies in sequential data. Similarly, the Gated Recurrent Unit (GRU) is a specialized type of RNN, simpler than an LSTM with fewer parameters and faster training but typically achieving similar results as LSTMs. [23] These are frequently used in hybrid models to complement the spatial feature extraction capabilities of CNNs.

Deep Belief Networks (DBNs), comprising layers of stacked restricted Boltzmann machines, were among the earlier deep learning architectures applied to EEG data. Although less commonly used in recent years, they have shown utility in unsupervised feature learning.

Autoencoders are particularly useful for dimensionality reduction and feature extraction, learning compact representations of EEG data, and are often used as a preprocessing step before classification.

Hybrid models combine multiple architectures such as CNN-LSTM hybrids. These models are designed to capture both spatial and temporal dynamics of EEG signals, thereby improving classification accuracy.

While CNNs and their hybrid variations dominated the reviewed studies, the lack of consensus on the optimal architecture underscores the exploratory nature of this field.

Moreover, it is important to note that this review was conducted prior to the advent of Transformers, a groundbreaking architecture that has revolutionized AI by beating state of the art performance in applications that capture long-range dependencies and attention-based modeling. Transformers have already demonstrated immense potential in domains beyond EEG analysis, and their application to sleep stage detection represents a promising area for future research. [24]

The diversity of architectures highlighted in the review emphasizes the ongoing evolution of deep learning methodologies for EEG-based sleep stage detection.

E. Summary of Research Gaps and Objectives

1) Identified Gaps: Despite advancements, the application of the ACT to EEG data remains limited, with few studies exploring its full potential. Similarly, the integration of advanced feature extraction methods with deep learning is still an emerging field.

There is a lack of studies integrating the ACT with deep learning for EEG analysis, particularly in sleep stage detection. The potential of the ACT for capturing subtle physiological changes has not been fully realized.

Furthermore, there is no consensus on the optimal deep learning architecture for sleep stage detection. The field of Machine Learning is extremely active, with novel architectures being developed on a weekly basis.

IV. Methods

A Python pipeline was developed for this thesis with the purpose of evaluating the Adaptive Chirplet Transform as a novel feature extraction method for EEG based sleep stage prediction.

Below follows a diagram of the full pipeline from raw EEG signal to sleep stage prediction:

A. Pre-Processing

As aforementioned, one of the key parts of any study using EEG data is to pre-process the EEG signals in an attempt to remove as much noise in the data as possible. For this thesis a Python pipeline for EEG data pre-processing was developed. The first step in

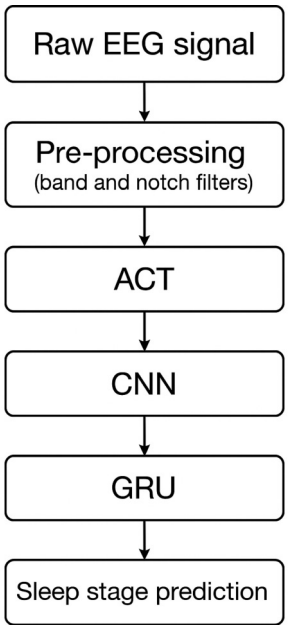


Fig. 4: Diagram of pipeline from raw EEG data to final sleep stage prediction

the pipeline is to download all of the data as an MNE raw data class. MNE is the most commonly used Python library for analysis of EEG and MEG data and it is used to implement the filters used in this pipeline. [25] The pre-processing pipeline includes bandpass filtering to isolate relevant frequency bands associated with sleep stages and notch filtering to eliminate powerline interference, a common artifact in EEG data. The bandpass filter between 1 to 40 Hz was applied using a zero-phase FIR filter with a Hamming window, followed by a notch filter at 50 Hz to eliminate power line interference; both filters were implemented using MNE-Python's `raw_data.filter` and `raw_data.notch_filter` functions with a `firwin` design. The firwin design was a chosen parameter which has been deemed ideal for EEG signal processing. From Widmann et Al. *Digital filter design for electrophysiological data - a pratical approach*: "FIR filters are easier to control, are always stable, have a well-defined passband, can be corrected to zero-phase without additional computations, and can be converted to minimum-phase. We therefore recommend FIR filters for most purposes in electrophysiological data analysis." [26]

These preprocessing steps are essential for ensuring that the EEG signals have some reduction from noise and artifacts while retaining the physiological features necessary for accurate sleep stage classification. The pipeline was designed to process EEG data in a consistent and scalable manner, facilitating future experimentation with both small-scale and large-scale datasets.

Another significant area of progress is the meticulous design of data segmentation and epoching strategies. Recognizing that the accurate detection of sleep state transitions requires both temporal precision and computational feasibility, epoch lengths were carefully chosen. These lengths were optimized to strike a balance between capturing rapid changes in brain activity and maintaining manageable computational loads for the ACT and subsequent analyses. This alignment ensures that the data preparation phase supports the broader objective of precise sleep stage detection.

1) Windowing and Overlap Strategy: Following the initial cleaning of the data, the EEG signal was segmented into overlapping epochs of 15 seconds. Windowing was applied to each segment to reduce spectral leakage, which are artifacts introduced by the assumption that the signal is periodic across window boundaries.

A variety of window functions were considered, each balancing trade-offs in main-lobe width, side-lobe attenuation, computational cost and ease of implementation.

The following windows were considered: Dirichlet, Bartlett, Hann, Hamming, Blackman and Kaiser. The **rectangular window** (Dirichlet window) applies no tapering, preserving amplitude but leading to significant spectral leakage due to high and slowly decaying side lobes. The **Bartlett window** (triangular) offers slightly reduced leakage but lower energy retention. **Hann** and **Hamming** windows are cosine-based functions commonly used in signal processing. Hamming has a slightly narrower main lobe and better side-lobe attenuation than Hann. The **Blackman window** provides even greater side-lobe suppression at the cost of a wider main lobe and increased computational load. Lastly, the **Kaiser window**, which is based on the Bessel function, offers flexible trade-offs by adjusting the shape parameter β.

For this thesis, the **Hamming window** was selected for its favorable balance of computational efficiency, main-lobe width, and side-lobe suppression. It is also the default window in MNE-Python's FIR filter implementation, which was used for the band-pass filter and notch filter that were applied in the previous step of the pipeline.

To mitigate the known issue of edge degradation, where signal quality is poorer at the edges of a window, a 75% overlap was used between consecutive epochs. This ensures that each time point is well represented in at least one central region of a window, where approximation quality is highest. The impact of windowing on signal quality is illustrated in Figure 5.

The image above shows just how much difference the application of a Hamming window made when compared to a Dirichlet window in terms od approximation accuracy. Using the same exact ACT parameters

TABLE I: Comparison of Common Window Functions

Window Type	Main-lobe Width	Stopband Attenuation	Features
Rectangular	Narrow	~13 dB	High leakage
Bartlett (Triangular)	Moderate	~26 dB	Simple, low leakage
Hann (Hanning)	Moderate	44 dB	Smooth cosine taper
Hamming	Moderate	53 dB	Better passband performance than Hann
Blackman	Wide	74 dB	Excellent attenuation, more computation
Kaiser ($\beta = 6$–8)	Adjustable	Flexible	Tunable trade-off via β

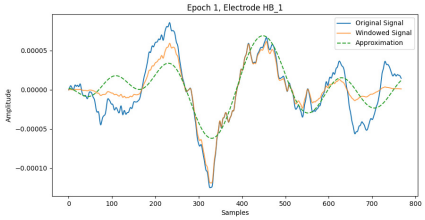


Fig. 5: Effect of Hamming Window on Signal Approximation

and ACT dictionary, the signal reconstruction under the Hamming window is visibly considerably superior.

B. Dataset

To further strengthen the study, the Bitbrain Open Acess Sleep Database was selected for analysis. [27] This dataset was chosen due to its high quality labeling procedure and abundant amount of sleep data, which provided enough samples to train a machine learning model on it. Providing a more reliable foundation for the thesis experiments and a better ground-truth to compare the model's accuracy against.

The dataset consists of 128 full nights of sleep data from 108 participants, including Polysomnography (PSG) data, EEG data, and a sleep stage label.

The sleep stage label included Wake, NonREM sleep stages one through three, REM sleep, PSG disconnection, and lastly a label for artifact or missing data.

Additionally, the labelling was done by three expert sleep scorers independently, and they followed the criteria developed by the Academy of Sleep Medicine. [28] Labels were provided for epochs of 30 seconds, and label had to be agreed upon by at least two of the scorers. In cases where a consensus was not achieved between the three experts, a fourth one was brought in to make the final decision. This was done to reduce possible human error and disagreement in sleep scoring, as there is inherent variability between experts when it comes to classifying sleep stages, with an estimated inter-scorer agreement of approximately 85%. [29]

The selected dataset was subjected to the preprocessing pipeline, downloading both EEG channels into an MNE object and then applying the band pass and notch filters, followed by data segmentation and epoching and the application of the ACT.

C. Adaptive Chirplet Transform

Building upon the theoretical background of the algorithm presented in the background, this section details how the ACT was implemented and adapted for use in this thesis.

1) Dictionary parameter tuning: The ACT Python open source implementation from Barghava was used as a baseline. [9] However, early testing, particularly informed by a conference paper published by our research group, revealed many changes were necessary to the initialization parameters for the ACT class and dictionary generation. [30]

The chirplet dictionary is created by generating Gaussian chirplets over the following user-defined ranges: $t_c \in [t_{\min}, t_{\max}]$ in steps of δt, $f_c \in [f_{\min}, f_{\max}]$ in steps of δf, $\log \Delta t \in [\log \Delta t_{\min}, \log \Delta t_{\max}]$ in steps of $\delta(\log \Delta t)$, and $c \in [c_{\min}, c_{\max}]$ in steps of δc.

Each point in this 4D parameter space corresponds to a unique chirplet, which is generated and stored as a row in a dictionary matrix. Given an input signal, the transform iteratively selects chirplets that best approximate the signal via dot product projection, followed by local parameter refinement using numerical optimization. The result is a compact, interpretable time-frequency representation consisting of a set of optimized chirplet parameters and their associated coefficients.

The defined parameter ranges have to be carefully selected for the given problem. For example, Bhargava used this implementation to study a P300 signal. The P300 is an event related potential which has a delay between stimulus and response of roughly 250 to 500ms. Thus, when using the ACT to analyze P300, the $[t_{\min}, t_{\max}]$ parameter may range between 0 to 500ms.

However, this time range would not work for sleep stage research, as discussed in background the scientific consensus for analyzing EEG data for sleep staging is 15 second epochs. This is roughly the amount of time

that shows enough transient features in the brainwave data such that the differences between sleep stages can be captured. With shorter epochs there might not be enough frequency changes and patterns that would allow an expert or machine learning model to properly figure out which sleep stage the epoch corresponds to.

Besides this, running the ACT on 500ms alone for sleep would require thirty times more epochs to be run, which would be computationally prohibitive for processing a full night of sleep.

When it came to the initialization parameters used for the ACT for this study there were two majors factors two consider. First and foremost was the fact that there was a significant trade-off between computational requirements and accuracy of signal approximation.

An incredibly near perfect signal approximation and reconstruction can be achieved with the Adaptive Chirplet Transform, but it requires a dictionary so extensive that the computation time is simply too large and the RAM requirements alone make it so that it can't be run on most modern laptops.

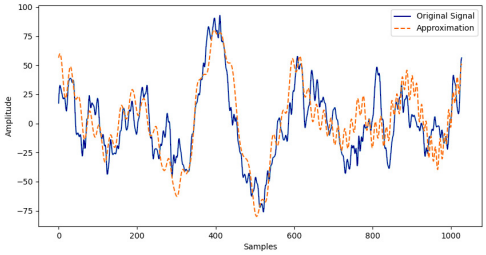
The second factor to consider was physiological. The time ranges should match the rough time duration where enough change has happened within the EEG signal such that sleep scoring could be achieved, which is roughly between 5 to 30 seconds. Similarly, the frequency range combined with the chirp rate should match known values of frequency ranges of brainwaves, typically listed as 0.5 to 40 Hz. These provide the bounds and guidelines on the minimum and maximum ranges of each of the parameters.

The step size, however, is purely motivated by the above mentioned trade off of computational requirements and overall signal quality. Figures below demonstrate the effect of varying step size and parameters on the accuracy of the approximation.

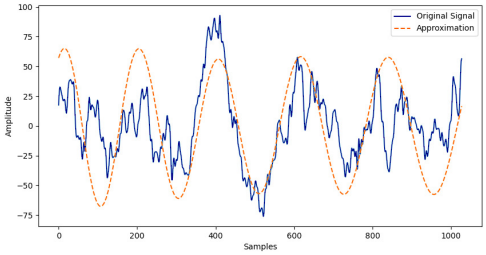
From the images below, it is clear to see that the smaller the step size, the better the approximation becomes. Note that the difference between 0.1 to 0.5 in frequency step size was the largest, where with 0.1 the reconstructed signal is a very close match of the original, especially in the middle, with some leakage from the sides causing a slightly less accurate reconstruction. This is in part due to the fact that the image above was constructed using a Dirichlet window, which has this known effect. The final parameters were constructed using a Hamming window which greatly minimizes the seen leakage effect at the boundary.

The visualization code that generated the graphs above was used to analyze epoch by epoch of several different parameter step sizes and after trial and error a final parameter range was decided upon. The final parameters are shown below:

Final Parameters: t_c: (0, 3840, 64); f_c: (0.6, 15, 0.2)

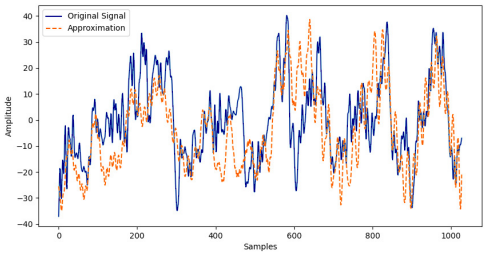


(a) 0.1

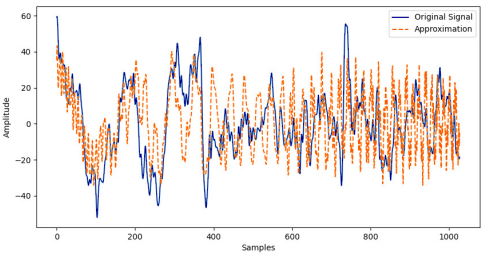


(b) 0.5

Fig. 6: Impact of the Frequency Step Size on Approximation of the ACT.



(a) 20



(b) 200

Fig. 7: Impact of the Time Step Size on the Approximation of the ACT

The time scale is in samples rather than seconds, since the sampling rate of the EEG recording device used was 256Hz, 3840 represents 15 seconds worth of sleep EEG data. A number that was chosen to follow the literature consensus on sleep stage analysis.

2) Parallelization of code: The original implementation of the Adaptive Chirplet Transform (ACT) was not well-suited for large-scale processing. It relied on CPU-based loops and NumPy operations, which made it impractical for analyzing full-length EEG recordings, especially when using longer epochs, high overlap, or finely-sampled parameter dictionaries. After reviewing performance limitations, I rewrote the pipeline to take advantage of GPU computing and parallel processing.

To run the transform more efficiently, the ACT code was adapted to work with CuPy, which mirrors the interface of NumPy but runs operations on the GPU. The new implementation was run on a Lambda Labs server equipped with an NVIDIA A100 SXM4 GPU with 40 GB of VRAM. The EEG data, along with the generated chirplet dictionary and intermediate arrays like window functions, were all transferred to GPU memory so that the ACT could be computed without moving data between devices during processing.

One of the main changes involved rewriting how epochs were created and processed. Instead of slicing each epoch on the CPU, the entire EEG signal was transferred to GPU memory at once. Epoch segmentation, windowing, and chirplet projections were then done in parallel on the GPU. The transform itself, especially the dictionary matching and optimization steps, saw major speedups due to the GPU's ability to handle thousands of operations at once. Only the final results, such as the fitted parameters and coefficients, were transferred back to CPU memory for saving and plotting on csv files. For reference, a csv file containing one full night of sleep was on average 900 Mega bytes long.

These changes made the ACT usable for full-night recordings and also opened the door to running more detailed parameter sweeps and real-time variants in future work. The entire GPU pipeline was integrated into a single script that could batch process subjects and automatically save results in a format suitable for later use in neural network training.

Despite these improvements, with the ACT dictionary parameters used, and the 75% window overlap, it still took 33.4 hours to run 3 out of the 128 full nights of sleep, costing a total of USD$43.11. With an average time of eleven hours of GPU time to process eight hours worth of EEG. The computational complexity of the Adaptive Chirplet Transform remains the biggest roadblock preventing it from having the capacity of becoming the standard EEG feature extraction method and achieving widespread usage in EEG data analysis and other domains and fields of signal processing.

Due to financial constraints and not having access to GPUs in house, this research was limited to only processing three out of the one hundred and twenty eight nights of sleep. Due to the sheer amount of epochs, in part due to the 75% overlap, three nights of data was still enough to provide over twelve thousand epochs. However, after processing it so it would be suitable for the Neural Network, it ended up resulting in 10,633 samples.

D. Deep Learning Based Sleep Stage Detection

This section details every step taken from preparing the data post ACT to making it suitable to be used as an input into the Neural Network, and the design and architecture of the Neural Network itself.

1) Input preparation for Neural Network: The implementation of the ACT used returns a csv file which contains per row, a list of coefficients (of length 5 since a 5^{th} order transform was employed), a list of parameters (of length 20), a list of residue, original signal, and error.

A separate Python pipeline was coded to handle the data preparation for the Neural Network, starting from downloading the csv files containing the results from the ACT.

The pipeline begins by extracting only the coefficients and parameters of each epoch, and transforming it into a 5x5 matrix. Where the rows of the matrix represent a each respective ACT order approximation. Such that each coefficient was joined with its respective function defining four parameters, and the first column of the matrix stood for the coefficients, wheres the remaining four columns were filled with the parameters.

This restructuring was done to make use of the power of two-dimensional Convolutional Neural Networks, which have been established as the most commonly used architectures for EEG data. Furthermore, this ordering made the structure of the data more explicit with the hopes it would be easier for the Neural Network to identify the most important features of the data faster.

For example, within this ACT resultant 5x5 matrix, the first column will always be more important than the second and so forth. This information could be given to the Neural Network with careful selection of initialization of weights.

Another crucial step with the data was to relabel it carefully. As previously mentioned, the original EEG data from Bitbrain contained labels per every 30 second epoch. However, since the Hamming window was applied with 75% overlap, the epochs no longer spanned only single labeled epochs. Rather, there was a large amount of epochs that spanned two different labels. To ensure no sample points were unnecessarily lost, an algorithm was written to calculate the exact time of

each new ACT processed epoch and map it back to the original labeled epochs. Then, the algorithm checked for every epoch that spanned two labels (due to the duration none spanned more than two), and checked whether the overlapped labels were the same or different. If the sample was overlapping a sleep stage transition period (meaning it spanned two different labels) it was discarded from the dataset.

Following this, epochs assigned to labels -2 (PSG disconnected), 3 (Non REM sleep stage 3), and 8 (artifact) were removed from the classification. This was because within the 3 subjects data alone there was essentially no data points for these labels (less than 0.1% of the data). This was not enough for the model to learn to distinguish these patterns but if included it made the learning process of the model much slower and had a significant negative effect on the performance and accuracy of the model. Regardless, two of the stages represent mal-labeled data, so removing it altogether from the dataset was an improvement in the overall data quality used for training the model and made it so that the model focused solely on the standard sleep stages, Wake, N1, N2, and REM.

The dataset was then split into training, validation, and test sets using a stratified splitting strategy. Stratification was done to preserve the original label distribution across each of the 4 categories, which was especially important because some categories had considerably more data points than others. The data was split such that 70% was used for training and the remaining 30% was split evenly between validation and training.

To mitigate the issue of massive class imbalance, the training data was oversampled to balance class count. The oversampling was done by duplicating samples for the underrepresented classes until all classes had the same number of examples. This ensures the model would not bias towards more frequent stages during training.

Oversampling improved the accuracy of the model considerably, almost doubled it in fact. However, it is only done for the training data and not for the validation or test. This is a common practice in the field of Machine Learning, as the validation and test are meant to be representative of how the model would generalize and perform with real world data, oversampling is generally avoided and frowned upon for validation and test sets.

After the oversampling, all feature tensors were standardized using StandardScalar from the sklearn library. This transforms the tensors such that they have a mean of zero and unit variance.

Lastly, the input tensors were grouped into small batches of a given sequence length. The sequence length used for this particular model was four, this value was a hyperparameter of the model. The purpose of these batches was so that temporal information could be passed into the model and to take advantage of the fact that typically sleep stages remain constant for much longer than the period of an individual epoch. Therefore, if one knows the sleep stages of a given amount of previous epochs, this information can be used to influence the decision of the model. If the previous three epochs were all at a given sleep stage, then the probability that the current epoch is at that same sleep stage should be higher.

These batches were shuffled before generating the validation/train/test splits to help with generalization of the model.

2) CNN + GRU Hybrid Model: To model the spatial and temporal structure of the chirplet-transformed EEG data, a hybrid neural network architecture combining Convolutional Neural Networks (CNNs) with Gated Recurrent Units (GRUs) was used.

The CNN component is applied independently to each time step in the input sequence. Each input matrix has shape 1×5×5 (channels × height × width), and is passed through a single convolutional layer with 16 output channels and a kernel size of 3×3 (with padding to preserve spatial dimensions). This layer captures local patterns between chirplet features while maintaining a manageable number of parameters. The resulting activation for each time step has shape 16×5×5, which is then flattened into a vector of size 400. These 400-dimensional vectors are stacked in temporal order, forming a [batch size, sequence length, 400] tensor passed to the GRU.

The GRU layer contains 2 stacked layers, each with a hidden size of 64. It processes the temporal sequence of CNN-extracted features in a recurrent manner, learning to model dependencies and transitions across time. The final hidden state from the GRU is used as a summary representation of the entire sequence. This final state is passed through a fully connected linear layer to produce class scores for sleep stage classification.

The CNN-GRU model was trained using cross-entropy loss with class weights to compensate for class imbalance, and optimized using the Adam optimizer. Class weights were computed from the training data after oversampling to maintain fairness across classes.

V. Results and Analysis

Below is the confusion matrix from running the CNNGRU model described above for 100 epochs, with a learning rate of 0.001, batch size of eight and sequence length of four on the Chirplet Transformed data from the first 3 nights of sleep of the Bitbrain Dataset.

The model achieved a final training accuracy of 91.54%, indicating that it was able to effectively learn the relationships between the input features and the sleep

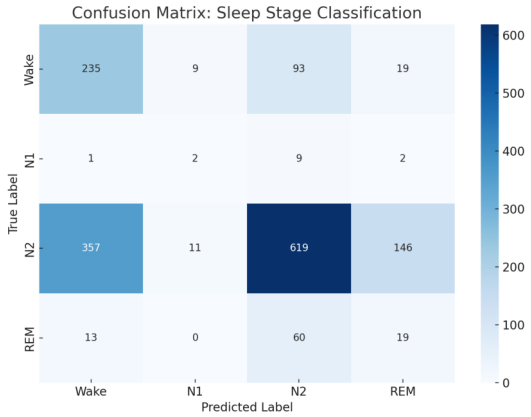


Fig. 8: Confusion Matrix - Final Model Result

stage labels during training. This high training performance suggests that the Adaptive Chirplet Transform (ACT) is a strong candidate for feature extraction in EEG-based sleep stage classification, thus validating the initial hypothesis of this thesis.

Since the ACT captures both time and frequency domain features with high precision, it allowed the CNN-GRU model to extract the most meaningful discriminative patterns for each sleep stage.

Furthermore, it does this by also achieving incredible compression as it approximates a 15 second epoch of raw signal containing 3840 sample points (stored as floats) using 25 floats instead, comprised of the chirplet coefficients and parameters.

The compression ratio (CR) is defined as the ratio of the original signal length to the compressed one:

$$\begin{aligned} \text{Compression Ratio} &= \frac{\text{Original size}}{\text{Compressed size}} \\ &= \frac{3840}{25} = 153.6 : 1 \end{aligned} \tag{2}$$

Equivalently, this corresponds to a compression rate of:

$$\begin{aligned} \text{Compression Rate} = \frac{25}{3840} \approx 0.00651 \quad \Rightarrow \\ 0.651\% \text{ of the original signal retained} \end{aligned} \tag{3}$$

This indicates that the ACT representation retains only about 0.651% of the original signal data per epoch, highlighting its effectiveness for compact, interpretable signal encoding.

However, the validation (52.76%) and test accuracy (57.93%) were significantly lower. This gap can be attributed to the fact that only the training data was oversampled to balance class distributions, while validation and test sets still had their original imbalances. The model may have generalized poorly to underrepresented classes it didn't see as frequently in validation/testing, highlighting the detrimental impact of class imbalance on final performance evaluation.

A closer look at the confusion matrix reveals important patterns. N2 sleep was the most prevalent class in the dataset (with 1133 test samples) and had the highest number of correct predictions (682), which contributed heavily to the overall accuracy. Wake was often confused with N2, as 115 of the Wake samples were misclassified as N2. REM sleep was also frequently confused with N2 (58 instances), while N1 showed poor classification performance overall, with only 1 correctly classified sample and most misclassified as N2.

This suggests a clear prediction bias toward N2, which is likely due to its overwhelming presence in the dataset. The network, trained on oversampled data but evaluated on imbalanced splits, defaulted toward the dominant class when uncertain.

Furthermore, while the overall test accuracy is lower than other results in the area of Deep learning categorization using EEG data, it is important to note that the particular issue of sleep staging is complicated and there is intervariability even amongst human experts, with results of about 85% agreement. This gives important context to the results.

VI. Conclusion and Future Work

The results achieved with this thesis show that the Adaptive Chirplet Transform has great potential to be an excellent method of feature extraction for EEG based Deep Learning applications. It manages to extract the most meaningful features of the data and it does so with astounding compression rate. A feature that could be very useful for Brain Chip Interface (BCI) applications where memory tends to be limited.

The relatively low accuracies displayed in the validation and test sets are to some extent explained by the large class imbalance in the data used. This can be likely vastly improved if the full pipeline is run on the full Bitbrain dataset, going through one hundred and twenty eight full nights of sleep rather than just three.

The largest roadblock at this time remains the computational complexity of the Adaptive Chirplet Transform. With the final ACT generation parameters used, the dictionary contains over 1.4 million Chirplet Gaussian functions, which makes it so that each epoch has to go through many dot products before finding the optimal chirplets. This currently takes around eleven hours to process eight hours worth of EEG data on A100 GPUs, making it almost but not quite suitable for real-time performance.

It is quite possible that this time complexity could be greatly reduced by the generation of a more optimized

dictionary. For example, instead of generating every single Chirplet Gaussian from the minimum and maximum values for the four parameters at the given step size, the dictionary could be crafted by the Chirplet Gaussian functions that are most widely seen and used. Further research is required to determine a reasonable dictionary size, and exactly how much this can speed up the ACT computation time. Once again, the optimal answer will undoubtedly come down to a trade-off between reconstruction accuracy and computational complexity.

A possible approach would be to run the current ACT version for a handful more hours' worth of data while keeping track of all of the chirplet functions that were chosen. Then create a dictionary solely from these chosen functions. This approach could also be done adaptively.

As for direct future steps for this research, given access to GPUs and the computational resources. The remainder of the Bitbrain dataset could be run with the current ACT implementation and the model could be trained with a much larger and more balanced dataset. This would test and hopefully confirm the hypothesis that the high training accuracy but low validation and test accuracies can be largely explained by the class imbalance in the current validation and test datasets.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all those who supported and contributed to this thesis.

I am especially grateful to my supervisor, Prof. Steve Mann. Your expertise and mentorship played a big role in shaping the direction and depth of this research.

A heartfelt thank-you goes to Nishant Kumar, whose support and technical knowledge were instrumental to this work.

Finally, I extend my appreciation to my friends and family for their continuous support.

REFERENCES

[1] L. Hu and Z. Zhang, *EEG signal processing and feature extraction*. Springer, 2019.

[2] J. Cui and W. Wong, "The adaptive chirplet transform and visual evoked potentials," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 7, pp. 1378–1384, 2006.

[3] S. Mann and S. Haykin, "Adaptive chirplet: an adaptive generalized wavelet-like transform," in *Adaptive Signal Processing* (S. Haykin, ed.), vol. 1565, pp. 402 – 413, International Society for Optics and Photonics, SPIE, 1991.

[4] Y. Jiang, W. Chen, M. Li, T. Zhang, and Y. You, "Synchroextracting chirplet transform-based epileptic seizures detection using eeg," *Biomedical Signal Processing and Control*, vol. 68, p. 102699, 2021.

[5] J. Cui and W. Wong, "Investigation of short-term changes in visual evoked potentials with windowed adaptive chirplet transform," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 4, pp. 1449–1454, 2008.

[6] M. Kaur, R. Upadhyay, and V. Kumar, "A hybrid deep learning framework using scaling-basis chirplet transform for motor imagery eeg recognition in brain–computer interface applications," *International Journal of Imaging Systems and Technology*, vol. 34, no. 4, p. e23127, 2024.

[7] C. López and K. J. Moore, "Enhanced adaptive linear chirplet transform for crossing frequency trajectories," *Journal of Sound and Vibration*, vol. 578, p. 118358, 2024.

[8] G. Zhang, Y. Cui, Y. Zhang, H. Cao, G. Zhou, H. Shu, D. Yao, Y. Xia, K. Chen, and D. Guo, "Computational exploration of dynamic mechanisms of steady state visual evoked potentials at the whole brain level," *NeuroImage*, vol. 237, p. 118166, 2021.

[9] A. Bhargava and S. Mann, "Adaptive chirplet transform-based machine learning for p300 brainwave classification," in *2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, pp. 62–67, 2021.

[10] S. S. Shinde and S. S. Kamthekar, "Person authentication using eeg signal that uses chirplet and svm," in *Computational Vision and Bio-Inspired Computing: ICCVBIC 2019*, pp. 338–344, Springer, 2020.

[11] J.-W. Qiu, J. K. Zao, P.-H. Wang, and Y.-H. Chou, "Consistent sparse representations of eeg erp and ica components based on wavelet and chirplet dictionaries," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 4014–4019, IEEE, 2010.

[12] Y. Lu, R. Demirli, G. Cardoso, and J. Saniie, "A successive parameter estimation algorithm for chirplet signal decomposition," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 53, no. 11, pp. 2121–2131, 2006.

[13] S. Motamedi-Fakhr, M. Moshrefi-Torbati, M. Hill, C. M. Hill, and P. R. White, "Signal processing techniques applied to human sleep eeg signals—a review," *Biomedical Signal Processing and Control*, vol. 10, pp. 21–33, 2014.

[14] X. Zhu, Z. Zhang, Z. Li, J. Gao, X. Huang, and G. Wen, "Multiple squeezes from adaptive chirplet transform," *Signal Processing*, vol. 163, pp. 26–40, 2019.

[15] C. K. Chui, Q. Jiang, L. Li, and J. Lu, "Analysis of a direct separation method based on adaptive chirplet transform for signals with crossover instantaneous frequencies," *Applied and Computational Harmonic Analysis*, vol. 62, pp. 24–40, 2023.

[16] Y. Qin, R. Yang, H. Shi, B. He, and Y. Mao, "Adaptive fast chirplet transform and its application into rolling bearing fault diagnosis under time-varying speed condition," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–12, 2023.

[17] M. Alaee and H. Amindavar, "Chirplet-based target recognition using radar technology," in *2008 5th IEEE Sensor Array and Multichannel Signal Processing Workshop*, pp. 451–454, 2008.

[18] Y. Jiang, W. Chen, M. Li, T. Zhang, and Y. You, "Synchroextracting chirplet transform-based epileptic seizures detection using eeg," *Biomedical Signal Processing and Control*, vol. 68, p. 102699, 2021.

[19] H. Altaheri, G. Muhammad, M. Alsulaiman, *et al.*, "Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: a review," *Neural Computing and Applications*, vol. 35, pp. 14681–14722, 2023.

[20] M.-P. Hosseini, A. Hosseini, and K. Ahi, "A review on machine learning for eeg signal processing in bioengineering," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 204–218, 2021.

[21] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert, "Deep learning-based electroencephalography analysis: a systematic review," *Journal of Neural Engineering*, vol. 16, no. 5, p. 051001, 2019.

[22] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015.

[23] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, Mar. 2020.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023.

[25] A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen, and M. Hämäläinen, "Meg and eeg data analysis with mne-python," *Frontiers in Neuroscience*, vol. 7, 2013.

[26] A. Widmann, E. Schröger, and B. Maess, "Digital filter design for electrophysiological data – a practical approach," *Journal of Neuroscience Methods*, vol. 250, pp. 34–46, 2015. Cutting-edge EEG Methods.

[27] E. López-Larraz, M. Sierra-Torralba, S. Clemente, G. Fierro, D. Oriol, J. Minguez, L. Montesano, and J. G. Klinzing, ""bitbrain open access sleep dataset"," 2025.

[28] R. Berry, S. Quan, and A. Abreu, *The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specifications*. Darien: American Academy of Sleep Medicine, version 2.6 ed., 2020.

[29] R. S. Rosenberg and S. Van Hout, "The american academy of sleep medicine inter-scorer reliability program: sleep stage scoring," *Journal of Clinical Sleep Medicine*, vol. 9, pp. 81–87, 2013.

[30] S. Mann, N. Kumar, J. P. Bicalho, M. Sibai, and C. Leaver-Preyra, "Adaptive chirplet transform-based sleep state detection," in *2025 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–6, 2025.

Early Validation of GPU-Accelerated Chirplet Transform

Nishant Kumar
Mann Lab
Toronto, Canada
nishant.kumar@mail.utoronto.ca

Steve Mann
Mann Lab
Toronto, Canada
mann@eyetap.org

***Abstract*—The Adaptive Chirplet Transform (ACT) offers a time-frequency decomposition for nonstationary signals, but its high computational cost has limited practical adoption. In this work, we present a preliminary GPU-accelerated implementation of ACT using CUDA unified memory and CuPy. On a 10-second EEG segment with a 324,000-element dictionary, our GPU implementation achieved a 12.9× speedup over a multithreaded CPU baseline (1.06s vs. 13.67s) using float32 precision. While dictionary generation remains faster on CPU, its one-time cost is mitigated by caching. Our results demonstrate the feasibility of using GPU acceleration for local, iterative workflows. A forthcoming conference paper will expand this work to include full performance profiling, resource usage analysis, and scalability benchmarking.**

Introduction

The increasing need for real-time and high-throughput analysis of signals demands more adaptive and efficient time-frequency analysis methods. Traditional approaches like the Fourier and wavelet transforms are limited in their ability to capture the nonstationary, transient features inherent in these signals. The Chirplet Transform, first introduced by Dr. Steve Mann, offers enhanced flexibility by allowing both frequency modulation and adaptive resolution, making it particularly well-suited for signal processing.

Despite its theoretical strengths, the practical adoption of the Adaptive Chirplet Transform (ACT) has been hindered by its computational demands. In a personal case study involving a sleep dataset with recordings from over 130 participants, an online GPU-based estimation (using vRAM-only memory models) projected a total runtime of over 42 days for full-scale Chirplet decomposition [1]. Worse still, any experimental misstep or configuration error would require restarting the process which makes cloud-only deployment both costly and inflexible. This showcases that the need for local GPU-accelerated tools to support iterative development.

To address this, we present a unified-memory-based GPU implementation of ACT that balances performance with flexibility. By enabling efficient local processing and debugging, our implementation is optimized for real-world research pipelines, including those involving large datasets.

Background and Prior Work

The Chirplet Transform is a time-frequency analysis method that extends the Gabor and wavelet transforms by having both time-varying frequency and frequency-modulated components. Originally developed in by Mann and Haykin in 1991, the Chirplet Transform provides a more adaptive representation of nonstationary signals, making it particularly useful for analyzing signals, where transient events and variable oscillations are common [2].

Recent years have seen increased demand for tools that can scale such transforms to large datasets and real-time pipelines. While initial implementations of the Adaptive Chirplet Transform (ACT) were CPU-based, the need for faster, more scalable analysis tools has led to early explorations of GPU acceleration.

In parallel, prior work on OpenVidia, an open-source GPU-accelerated signal processing library, demonstrated the feasibility of porting complex DSP algorithms to GPUs using CUDA [3]. Building on that philosophy, this project explores a dedicated GPU implementation of ACT, with optimizations that include unified memory, stream-based parallelism, and profiling-informed memory management. This represents a natural and necessary step in making Chirplet-based analysis practical for real-time and embedded use cases.

Methodology

We implement a GPU-accelerated version of the Adaptive Chirplet Transform (ACT) using CuPy and CUDA unified memory to enable efficient decomposition of signals. Unified memory allows seamless access between CPU and GPU, simplifying development and supporting large dictionary sizes without explicit memory transfer bottlenecks and increasing the amount of RAM that the ACT can use.

A four-dimensional chirplet dictionary is generated over time center, frequency, log-duration, and chirp rate parameters. Each combination defines a basis function used to analyze overlapping windows of the input signal. Dictionary generation and signal decomposition are parallelized: each GPU thread processes a unique parameter combination, enabling large-scale evaluation in a single pass. During transformation, the signal is iteratively approximated by projecting onto the dictionary, selecting the best-matching chirplet, optimizing its parameters, and subtracting it from the signal. Intermediate results are cached to support debugging and reuse. This approach balances performance and flexibility, making ACT viable for real-time or iterative analysis on large datasets.

Results and Discussion

In this preliminary evaluation, we focused on the transform phase of the Adaptive Chirplet Transform, as the dictionary can be generated once and cached. This stage benefits significantly from GPU acceleration: for a dictionary of approximately 324,000 chirplets, the GPU completed the transform in 1.06 seconds, compared to 13.67 seconds on a multithreaded CPU baseline which is approximately a 12.9× speedup. All tests were conducted using float32 precision.

While dictionary generation remains faster on CPU (21 seconds) than GPU (107 seconds), this step is performed only once per parameter set and is therefore less critical in interactive or iterative workflows, and it can also be cached to use for other datasets. The transform was performed with an order of 5, meaning five chirplets were used to approximate each signal segment. Each test used a 10-second EEG epoch sampled at 256 Hz. Parameter ranges were as follows:

- Time center: 0 to 2560 (step size: 32 samples)
- Frequency: 0.6 to 15 Hz (step size: 1 Hz)
- Log duration: –4 to 0 (step size: 0.4)
- Chirp rate: –10 to 10 (step size: 0.75)

These results demonstrate that GPU acceleration of the Chirplet Transform's search phase is both feasible and practical for real-time or iterative signal analysis. Despite slower dictionary generation on GPU, the ability to cache and reuse the dictionary enables rapid decomposition, making the approach well-suited for local workflows such as signal tuning, prototyping, and personalized adaptation.

A more comprehensive performance profile which will cover runtime breakdowns, resource usage, and scalability will be presented in a forthcoming conference paper.

References

[1] S. Mann, N. Kumar, J. P. Bicalho, M. Sibai, and C. Leaver-Preyra, "Adaptive chirplet transform-based sleep state detection," in *2025 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–6, 2025.

[2] S. Mann and S. Haykin, "Adaptive chirplet: an adaptive generalized wavelet-like transform," in *Adaptive Signal Processing* (S. Haykin, ed.), vol. 1565, pp. 402 – 413, International Society for Optics and Photonics, SPIE, 1991.

[3] J. Fung and S. Mann, "Openvidia: parallel gpu computer vision," in *Proceedings of the 13th Annual ACM International Conference on Multimedia*, MULTIMEDIA '05, (New York, NY, USA), p. 849–852, Association for Computing Machinery, 2005.

Interactive Quaternion Visualization and Form Training Using an IMU and LED Strip

Anne Chow, Matthew Maciesowicz, Danilo Zelenovic
University of Toronto, Faculty of Applied Science and Engineering
Email: anne.chow@mail.utoronto.ca, m.maciesowicz@mail.utoronto.ca, danilo.zelenovic@mail.utoronto.ca

***Abstract*—This project presents an interactive game system designed to teach the concept of quaternions and enhance exercises using an inertial measurement unit (IMU) and an addressable LED strip. The system, mounted on a balance board, visualizes quaternion-based rotations in order for this concept to be more understandable and implements a gamified "water spilling" simulation to encourage controlled movement, leading to better form, increased effectiveness of workouts, and increasing the fun simultaneously. The ESP32 microcontroller processes IMU data, translates orientation into LED patterns, and dynamically adjusts visuals based on user movement. This report details the system's design, implementation, and experimental evaluation, highlighting its effectiveness in education, fitness, and gaming applications.**

***Index Terms*—Quaternions, Fitness, IMU, Addressable LED Strip, ESP32 Microcontroller, Gamification, Motion tracking.**

I. Introduction

Understanding quaternions is crucial for modern motion-tracking applications, yet their abstract mathematical nature makes them difficult to grasp intuitively. In addition, balance training is a key component in sports and rehabilitation. This project integrates these domains by providing an interactive real-time visualization of quaternions while simultaneously engaging users in a gamified exercise.

The system consists of an ESP32DevkitV1 microcontroller, an MPU9250 IMU, and a WS2812 addressable LED strip. The IMU captures orientation data, which are then converted into quaternion-based visualizations on the LED strip. Furthermore, a balance-based "water spilling" game provides real-time feedback based on the user's stability. As the balance board tilts, the user will begin losing "water", and they have failed their game once all of the water is lost. Quaternions are a generalization of complex numbers, consisting of a scalar and a vector part, formed by 3 complex numbers. These quaternions are mainly used in the fields of computer graphics or robotics to represent and manipulate rotations in a 3-dimensional space. Traditionally, Euler angles have been used. However, quaternion-based measurements provide many useful benefits. These include the fact that quaternions avoid the gimbal lock problem, which can occur when using Euler angles. When a gimbal lock occurs, 2 axes align and a degree of freedom is lost. Additionally, the quaternion approach also offers a more compact representation of the rotations, allows for simpler interpolation, and is not affected by drifting values as often, which can present a large problem when using yaw values, for example. By combining the exercise and game portions, which aim to improve training results by correcting form, we are able to teach rotational geometry by allowing users to feel as if they themselves have become quaternions within a pool of water.

II. Related Work

Quaternions are widely used in robotics, aerospace, and augmented reality due to their ability to represent rotations without gimbal lock [1]. Various projects have demonstrated quaternion visualizations [2], but few integrate them into a tangible, interactive system. Similarly, balance training has been improved using digital tools [3], but the use of LED-based feedback in gamified formats remains relatively unexplored. Dr. Mann has the idea of creating a quaternion ball game in which users can learn to become quaternions while potentially rehabilitating the ankles or playing various games, and this concept was adapted to this created fitness training device. Balance boards have mainly been used for the purpose of rehabilitation. However, they have been expanded and benefits have been shown when used for other purposes such as conditioning or strengthening. They can improve balance, posture, prevent injuries, increase core strength, and more. Keeping balance requires 3 main bodily functions. The visual, vestibular, and proprioceptive systems. Balance boards mainly help train the proprioceptive system, which keep track of your position in 3 dimensional space, similarly as the quaternion display may do. While wobble boards currently do exist, none seem to combine games along with them, which is where the innovation of our project excels. By incorporating the addressable LEDs in a sequential wave imprinting machine (SWIM) fashion with the moving lights, it allows for even more creative gaming options while exercising.

III. Math Behind Quaternions

Quaternions are a mathematical construct used to represent rotations in three-dimensional space. A quaternion q is defined as:

$$q = w + xi + yj + zk \tag{1}$$

where: i) w is the scalar part of the quaternion, ii) x, y, z are the components of the vector part of the quaternion, iii) i, j, k are the fundamental quaternion units.

The quaternion multiplication operation is used to combine rotations. The product of two quaternions $q_1 = w_1 + x_1 i + y_1 j + z_1 k$ and $q_2 = w_2 + x_2 i + y_2 j + z_2 k$ is given by:

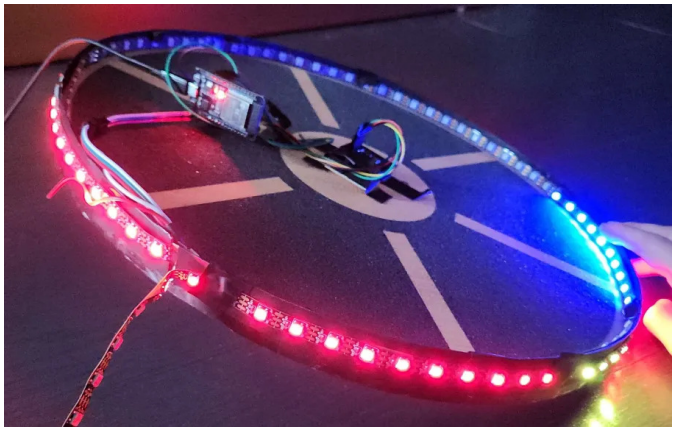


Fig. 1. Board tilted with only few remaining LEDs for water game

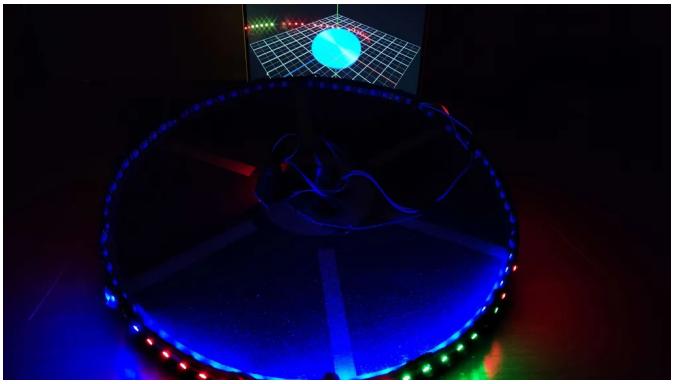


Fig. 2. Water spilling towards tilt and screen mirroring movements

$$\begin{aligned} q_3 = q_1 \cdot q_2 = &(w_1 w_2 - x_1 x_2 - y_1 y_2 - z_1 z_2) \\ &+ (w_1 x_2 + x_1 w_2 + y_1 z_2 - z_1 y_2)\, i \\ &+ (w_1 y_2 - x_1 z_2 + y_1 w_2 + z_1 x_2)\, j \\ &+ (w_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 w_2)\, k \end{aligned} \quad (2)$$

The quaternion normalization process is also essential when handling rotations. A normalized quaternion q_{norm} is given by:

$$q_{\text{norm}} = \frac{q}{|q|} \quad (3)$$

$$i^2 = j^2 = k^2 = ijk = -1 \quad (4)$$

where $|q| = \sqrt{w^2 + x^2 + y^2 + z^2}$ is the norm (magnitude) of the quaternion.

In our application, the IMU data (accelerometer and gyroscope readings) are used to compute the rotation matrix, which is then converted into quaternions for more stable and efficient calculations. This is done using the following formula for a rotation matrix R and its corresponding quaternion q:

$$\begin{aligned} q = &\frac{1}{2}\sqrt{1 + R_{00} + R_{11} + R_{22}} \\ &+ \left(\frac{R_{21} - R_{12}}{4}, \frac{R_{02} - R_{20}}{4}, \frac{R_{10} - R_{01}}{4} \right) \end{aligned} \quad (5)$$

IV. System Design

The system incorporates multiple hardware/physical components, listed below.

A. Hardware Components

The system consists of:

- **ESP32**: Handles data processing and LED control.
- **MPU9250 IMU**: Provides quaternion data representing board orientation.
- **WS2812 LED Strip**: Displays quaternion data and water game effects.
- **Balance Board**: Platform for user interaction.
- **Power Supply**: A battery or external power source to drive the ESP32 and LED strip.

Fig. 3. Water tilting with quaternions displayed on the outer edge

These items are assembled in a manner that provides access to the board, while keeping the electronic components secured. The IMU is placed in the center of the board, to allow all directions of tilt to be mapped correctly and as accurately as possible. The LED strip is placed along the edge of the board. Along the inner edge is displayed the water game, which will light the board in blue initially, giving the appearance of a full bucket of water for instance. The outer edge contains the quaternion information, providing information on the direction of the board, and changing dynamically as the board adjusts. By slightly modifying the board and providing both an inner and outer edge, some portion of the LEDs can be seen from any possible angle that a person is able to reasonably look from. A computer screen is also added to allow more options, more visibility and clear visualization of the board as well, as it will wirelessly receive IMU data using the microcontroller, over WiFi.

B. Software Implementation

The software processes IMU data, applies quaternion mathematics, and updates the LED display. The game logic reduces LED availability based on tilt magnitude, simulating

water loss. The ESP32 controls the flow of the game and user interactions, with feedback mechanisms provided through the LED strip. The quaternion information is also directly displayed on the LED as well, with components shrinking or growing in magnitude, as to showcase them adjusting in real-time as the board is tilted or moved to be able to grasp the concept better. The software was written in the Arduino IDE, created with the specific hardware connections in mind. The FastLED library was used to control each LED separately, in real-time, based on the data received from the IMU. The code is adjustable, where the number of LEDs (or the amount of water) allowed for the water game can be adjusted based on the size of the board very easily by changing the definition. Global variables are used to store initial readings and game state values. Additionally, the amount of LEDs lost needs to be tracked. Once all of them are lost, then the game over state is displayed on the LEDs, where the game portion turns red, while the rest of the LED strip turns off. A function is created to begin reading data from the IMU, which is then converted to usable units. The MPU9250 register is used to have information written and read. The data from the IMU is written to this register, and then read by the microcontroller to be used in calculations. This value updates in real-time. Calculations are done to mitigate external factors such as noise or gravity, then once those are complete, the quaternion information is calculated based on the mathematical properties and formulas from the previous section of the report. I2C communication was configured between the IMU and the microcontroller to allow the devices to communicate smoothly using the SDA and SLC header pins. The values were printed first to the serial monitor for testing and debugging purposes. Additionally, WebSocket programming was implemented to send quaternion data and have it plotted to a screen as well. This allows for additional uses or visibility to users or bystanders on the current state of the activity. Wifi connections need to be enabled, and the credentials must be entered prior to start.

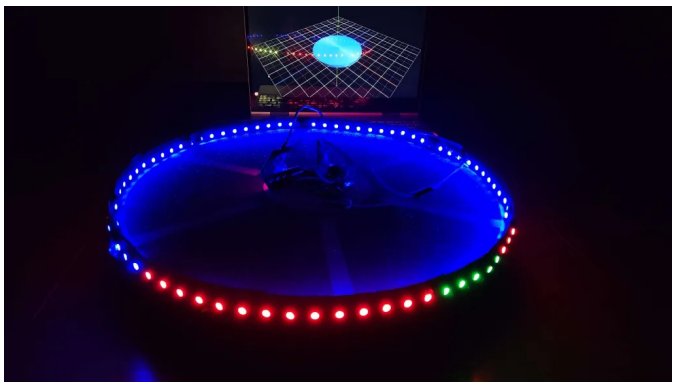


Fig. 4. Flat board with full water display

V. Implementation

The core implementation consists of the following steps:

1) **IMU Data Processing**: The ESP32 reads raw acceleration and gyroscope data, converts it into quaternions, and filters noise.
2) **Quaternion Visualization**: The LED strip colour-codes quaternion components to show orientation changes, representing each quaternion component separately.
3) **Water Game**: If tilt exceeds a threshold, LEDs begin to turn off, simulating water loss. The game resets when all water is lost. The game is automatically restarted once all water is lost from the portion of the LED strip provided, after the LEDs flash red to ensure users are aware that the game is over.

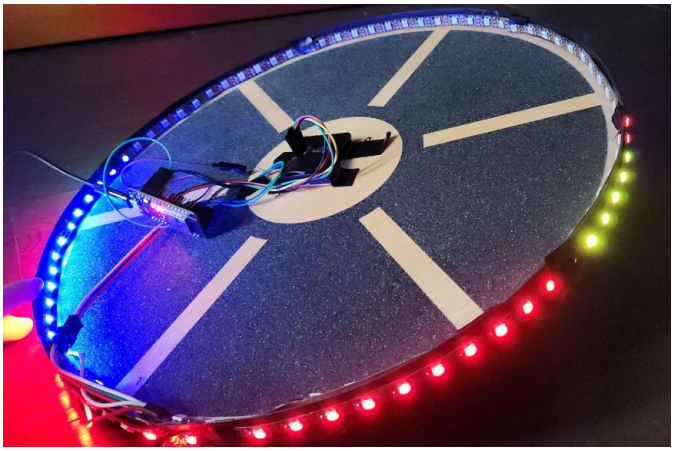


Fig. 5. Water tilting in inner portion with quaternions displayed on the outer portion

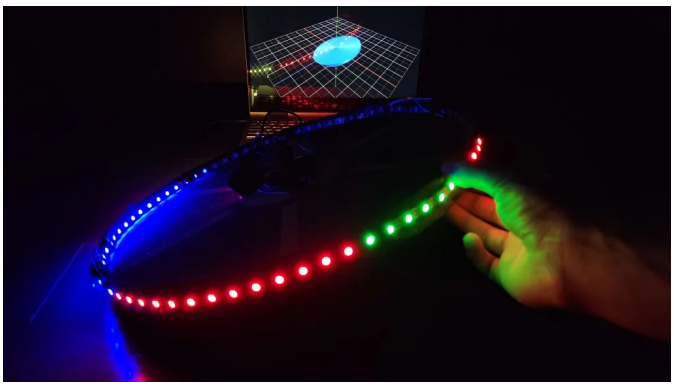


Fig. 6. Water moving towards tilt with monitor replicating movements

VI. Results and Discussion

Experiments showed that quaternion visualization effectively depicted orientation changes in real time. The water game provided an engaging way to encourage controlled movements. Challenges included sensor drift and LED update latency, which were mitigated through calibration and

optimized rendering algorithms. In addition, many provided libraries no longer functioned, meaning the math had to be done manually in order to compute the information required. The game kept users engaged, motivating them to maintain better balance while learning about quaternion-based motion. Users felt an increase in difficulty in balancing on the board as they utilized more muscles and groups they had not thought they required nearly as much. Users became more occupied and spent longer exercising than on average, as the additional challenge of not spilling water and the human nature of being competitive made users attempt to spill the least amount of water for as long as possible.

VII. Conclusion and Future Work

This project successfully demonstrates an interactive tool for visualization of quaternions and balance training. Future work includes integrating a scoring system, adding machine learning-based movement analysis to assess performance, and enhancing graphical feedback through additional sensors such as pressure sensors on the balance board to detect tilt more precisely. The users became not only more interested in working out, but also noted that they felt the exercise was higher quality than their typical ones. Something that can be done in the future is to implement the muse headband to take baseline workout or activity readings and compare them with the readings from our board, to achieve more accurate readings. Additionally, this can be implemented with various other workout equipment in order to help users improve form and allow them to combine bun with exercise. This project was successful, as it was able to meet the requirements tasked to it, and showed not only to function, but to increase user retention and ability.

References

[1] J. B. Kuipers, "Quaternions and Rotation Sequences," Princeton University Press, 1999.

[2] R. Dantzig, "Quaternion-Based Motion Control for VR Applications," IEEE VR Conference, 2012.

[3] M. Smith et al., "Digital Balance Training Systems," Journal of Biomechanics, vol. 45, no. 3, 2018.

[4] D. E. Garcia, K. W. Zheng, Y. Liu, Y. S. Tao, and S. Mann, "Painting with the Eye: Understanding The Visual Field of the Human Eye with SSVEP," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 3689–3694, 2020. DOI: https://doi.org/10.1109/SMC42975.2020.9283476.

[5] W. R. Hamilton, "Theory of Quaternions," *Proceedings of the Royal Irish Academy (1836-1869)*, vol. 3, pp. 1–16, 1844. Available: http://www.jstor.org/stable/20489494.

[6] W. R. Hamilton, *Elements of Quaternions*, Longmans, Green, and Co., London, 1866.

[7] S. Mann, "Phenomenological Augmented Reality with the Sequential Wave Imprinting Machine (SWIM)," in *2018 IEEE Games, Entertainment, Media Conference (GEM)*, pp. 1–9, 2018. DOI: https://doi.org/10.1109/GEM.2018.8516502.

[8] R. Mukundan, "Quaternions," in *Advanced Methods in Computer Graphics*, Springer London, pp. 77–112, 2012. DOI: https://doi.org/10.1007/978-1-4471-2340-8_5.

[9] G. Pratticò, F. De Lorenzis, and F. Lamberti, "Look at It This Way: A Comparison of Metaphors for Directing the User's Gaze in eXtended Reality Training Systems," in *2021 7th International Conference of the Immersive Learning Research Network (iLRN)*, IEEE, pp. 1–8, 2021.

[10] J. Rowberg, *MPU-9250/I2Cdevlib*, 2019. Available: https://github.com/jrowberg/i2cdevlib.

[11] S. Mann, J. Fung, R. Janzen, and M. A. Schneider, "Intelligent Image Processing (IIP): 'WearCam' and the 'MannFit Test' for Computer-Based Fitness Evaluation," in *2014 IEEE Games, Entertainment, and Media Conference (GEM)*, pp. 270–272. Available: http://wearcam.org/mannfit/IEEE_GEM2014_MannEtal_pages270-272.pdf.

Wireless Motion Sensing and Visualization using IMU and SWIM

Xu Ji, Mao Fan, ChuanZhi Li
Department of Electric and Computer Engineering
University of Toronto

***Abstract*—This paper presents a dual-system approach for motion sensing applications. First, a wireless motion-sensing wand is implemented using MPU9250 IMU and WS2812B LED strip, communicating via Bluetooth Low Energy (BLE) to a central processor for gesture recognition and LED animation control. Second, a motion-controlled bullet-dodging game is developed using ESP32-WROVER with MPU6050 IMU, employing Kalman filtering for attitude estimation. Both systems demonstrate real-time motion tracking capabilities with Kalman-filtered sensor fusion, providing responsive control for interactive applications. Experimental results show effective motion recognition and low-latency feedback in both implementations.**

***Index Terms*—Motion sensing, IMU, Kalman filter, BLE, gesture recognition, sensor fusion, embedded systems**

I. Introduction

We present two complementary motion-sensing systems: (1) a wireless gesture-controlled wand with LED feedback, and (2) a motion-controlled bullet-dodging game. Both implementations utilize inertial measurement units (IMUs) with Kalman filtering for accurate motion tracking.

II. Magic Wand Design

A. System Overview

We created a wireless motion-sensing wand using MPU9250 IMU and WS2812B LED strip. The IMU communicates over BLE to a PC, which processes orientation data and transmits commands to a Raspberry Pi Pico for LED animations.

B. Motion Estimation via Kalman Filtering

The MPU9250 provides acceleration and orientation data. The PC implements a Kalman filter for 3D position and velocity estimation.

1) Sensor Fusion: Accelerometer readings transformed to world coordinates:

```
def transform_acc_with_gravity(ax, ay, az, roll,
    pitch, yaw):
    # Convert sensor acceleration to world frame
    # using rotation matrices
    ...
    acc_avg = [sum(x[i] for x in acc_buffer) / len(
    acc_buffer)
               for i in range(3)]
```

2) Zero Velocity Update (ZUPT): ZUPT detects low-motion states:

```
if magnitude(acc_avg) < zupt_threshold:
            zupt_counter += 1
if zupt_counter >= zupt_stable_frames:
            zupt_active = True
```

3) Kalman Position Estimation: Position update equations:

$$\text{position}[i] += \text{velocity}[i] \times \Delta t$$
$$\text{velocity}[i] += \text{acc\_avg}[i] \times \Delta t$$

Correction step:

$$K = \frac{P}{P+R}, \quad \text{position}[i] += K \times (z - \text{position}[i])$$

C. Gesture Recognition Logic

Gesture mapping dictionary:

```
COMBOS = { ("U", "D"): 1, ("D", "U"): 2, ..., ("R",
    "D"): 12 }
```

Control command transmission:

```
await client_pico.write_gatt_char(PICO_WRITE_UUID, f
    "{code} ".encode())
```

D. Light Output on Pico

LED control function:

```
def control_lights(code):
    if code == "1": wave((0, 0, 255))   # Blue
    stream
    elif code == "4": rainbow_cycle(5)
    elif code == "12": blink((255, 0, 255))
```

E. Bluetooth Setup

The Pico advertises its BLE service with UUID 0x7777, while the PC connects to both WT901 and Pico:

```
ble = bluetooth.BLE()
ble.active(True)
ble.gatts_register_services(...)
```

Connection events and message reception are managed with IRQ as seen in the code below:

```
    def bt_irq(event, data):
        if event ==IRQ_GATTS_WRITE:
                msg = ble.gatts_read(rx_handle).
    decode().strip()
                control_lights(msg)
```

The results of this can be seen in Figure 1

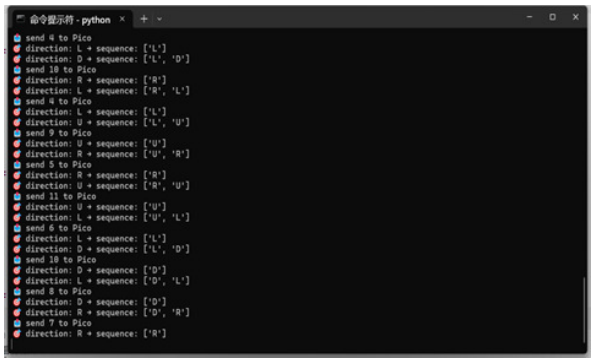


Fig. 1: Console output showing real-time direction recognition

III. Game Design — Motion-Controlled Bullet-Dodging Game

This experiment focuses on the development of a bullet-dodging game controlled entirely through motion, using data from an Inertial Measurement Unit (IMU). Specifically, we employ the ESP32-WROVER microcontroller to interface with an MPU6050 sensor, transmitting gyroscope and accelerometer data via serial communication to a Python-based game interface. Within the game, the player controls a blue square that must avoid red bullets while collecting green squares to accumulate points. This setup serves both as a test of real-time motion control and as an engaging interactive application for embedded sensor data processing.

IV. Attitude Estimation

A. Sensor Initialization

The MPU6050 IMU captures both linear acceleration and angular velocity data through the function call:

```
mpu.getEvent(&a, &g, &temp);
```

Before beginning real-time operation, the system collects 2000 samples to estimate zero offsets for the accelerometer and gyroscope along each axis. These offsets (e.g., `ax_offset`, `ay_offset`) are subtracted from subsequent sensor readings to minimize drift and bias errors in measurement.

B. Computing Time Difference

Accurate integration of angular velocity requires precise timing. The time difference Δt between sensor updates is calculated as:

$$\Delta t = \frac{\texttt{now} - \texttt{last\_time}}{1000.0}$$

This provides the integration timestep in seconds, allowing consistent angular updates across variable frame rates.

C. Estimating Angles from the Gyroscope

Using the gyroscope's angular velocity data, we estimate rotational angles as follows:

$$\begin{aligned} v_{\text{roll}} &= g.\text{gyro.x} \\ v_{\text{pitch}} &= g.\text{gyro.y} \\ \text{gyro\_roll} &= \text{previous\_roll} + \Delta t \cdot v_{\text{roll}} \\ \text{gyro\_pitch} &= \text{previous\_pitch} + \Delta t \cdot v_{\text{pitch}} \end{aligned}$$

These values accumulate over time to provide an estimate of the current orientation; however, they are prone to drift without correction.

D. Estimating Angles from the Accelerometer

To correct for long-term drift, we compute independent estimates of roll and pitch using accelerometer data and trigonometric relationships with gravity:

$$\begin{aligned} \text{acc}_{\text{roll}} &= \operatorname{atan2}\left(y - a_{y_{\text{offset}}}, z\right) \cdot \text{RAD\_TO\_DEG} \\ \text{acc}_{\text{pitch}} &= -\operatorname{atan2}\left(x - a_{x_{\text{offset}}}, \sqrt{(y - a_{y_{\text{offset}}})^2 + z^2}\right) \cdot \text{RAD\_TO\_DEG} \end{aligned}$$

These accelerometer-based angles act as a reference for correcting the integrated gyro readings.

V. Kalman Filter for Attitude Estimation

To fuse gyroscope and accelerometer data into a stable and drift-free orientation estimate, we implement a Kalman filter with the following steps:

A. Prediction Step

The error covariance matrix P is updated to reflect increased uncertainty over time:

$$P[0][0] += 0.0025$$

B. Kalman Gain Computation

The Kalman gain K is calculated to determine the weighting between the predicted and measured states:

$$\begin{aligned} K[0][0] &= \frac{P[0][0]}{P[0][0] + 0.3} \\ K[1][1] &= \frac{P[1][1]}{P[1][1] + 0.3} \end{aligned}$$

C. State Update

Filtered roll and pitch angles are computed by correcting the gyro estimates using accelerometer measurements:

$$\begin{aligned} \text{roll} &= \text{gyro\_roll} + K[0][0] \cdot (\text{acc\_roll} - \text{gyro\_roll}) \\ \text{pitch} &= \text{gyro\_pitch} + K[1][1] \cdot (\text{acc\_pitch} - \text{gyro\_pitch}) \end{aligned}$$

D. Covariance Update

Finally, the error covariance is updated to reflect the reduced uncertainty:

$$P[0][0] = (1 - K[0][0]) \cdot P[0][0]$$
$$P[1][1] = (1 - K[1][1]) \cdot P[1][1]$$

This recursive filtering approach ensures that orientation tracking remains smooth, accurate, and responsive throughout gameplay.

VI. Game Rules and Logic

A. Player Control

The player is represented by a blue square, whose position is updated in real-time based on the processed IMU data. Tilting or moving the sensor alters the square's position on screen, creating an intuitive motion-based interface as seen in Figure 2. The game's interface will now be explained and can be seen in Figure 3

B. Hazards: Red Bullets

Red bullets serve as dynamic hazards within the game:

- Bullets move unpredictably, bouncing around the screen.
- Colliding with a red bullet results in a penalty of 100 points.
- Upon collision, the bullet disappears from play.

C. Score Items: Green Squares

Green squares act as score boosters and gameplay triggers:

- Collecting a green square adds 100 points to the player's score.
- Each green square collected also causes a new red bullet to spawn, increasing the game's difficulty.

D. Game Duration and End Conditions

The game runs for a fixed duration of 60 seconds. After time expires, the final score is displayed, summarizing the player's performance in terms of agility and control precision.

VII. Serial Data Processing

A. Data Format

The sensor outputs data over a serial connection in the form of a string formatted as `num1/num2`, where `num1` and `num2` correspond to measurements along the vertical (Y) and horizontal (X) axes, respectively. Both values are constrained within a range of $[-70, 70]$, representing the bounded output of the sensor's motion or orientation tracking system.

B. Implementation Notes

To receive the data stream, a Python script utilizing the `pyserial` library listens on port `COM3`. The incoming string is parsed in real time to extract the vertical and horizontal components. These values are then used to control on-screen elements rendered via the `Pygame` library, which provides a 2D graphical interface and maps input dynamics to visual feedback. This setup enables a responsive system for visualizing and interpreting sensor data in an interactive environment.

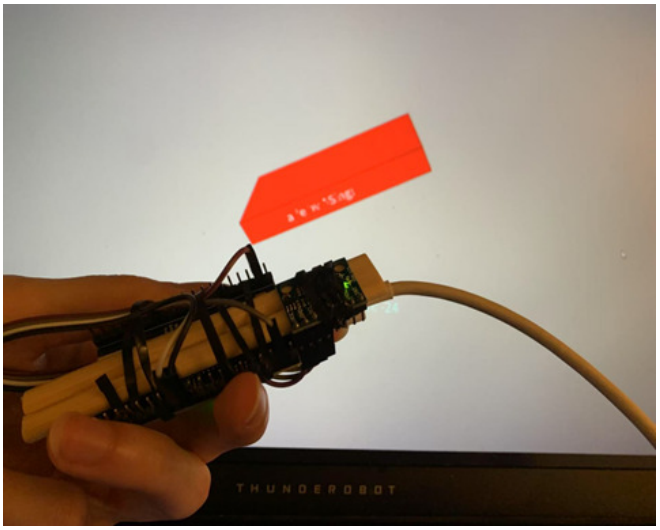


Fig. 2: Demonstration of real-time pose estimation using Kalman-filtered orientation angles to control player movement.

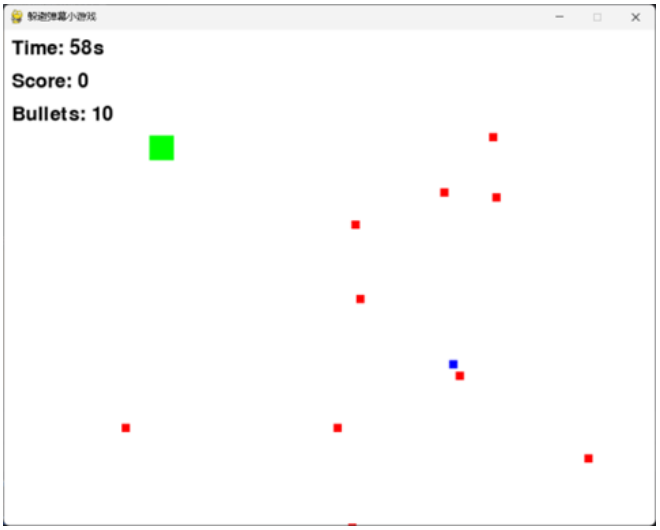


Fig. 3: Game window during play showing blue player square, red bullets, and green point icons

VIII. Results

Both systems demonstrated real-time performance with low latency. The gesture recognition system achieved high accuracy in controlled tests, while the game maintained stable frame rates during gameplay. Kalman filtering significantly reduced drift compared to raw sensor data, enhancing overall motion tracking precision.

IX. Conclusion

We have implemented two complementary motion-sensing systems using IMUs with Kalman filtering. The wireless wand provides intuitive gesture control with visual feedback, while the motion-controlled game demonstrates responsive interac-

tion. Future work includes multi-sensor fusion and machine learning-based gesture recognition.

REFERENCES

[1] , "MPU6050" [Video]. https://www.bilibili.com/video/BV1sL411F7fu, accessed May 2025.

[2] P. Zhou, M. Li, and G. Shen, "Use It Free: Instantly Knowing Your Phone Attitude," in *Proc. MobiCom*, 2014, pp. 605-616.

[3] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, pp. 35-45, 1960.

[4] S. Gomez et al., "Bluetooth Low Energy for Motion Tracking," *IEEE Sensors Journal*, vol. 18, no. 4, pp. 1595-1603, Feb. 2018.

Project Repository:
https://gitlab.com/XuLeoJi/ece516/-/tree/main/TheMagicWand

Correlating Brain Waves in Invironment with Emotional States in the Environment

UNIVERSITY OF TORONTO

Tanish Sharma

Objective

Analyze the correlation between brainwave patterns detected by Muse S electrodes and facial emotions captured by web-cam.

Investigate how environmental factors influence emotional states, incorporating concepts of invironment and environment (S. Mann).

Develop insights into what external and internal stimuli contribute to happiness or sadness.

Introduction

- Invironment (internal states) and environment (external surroundings) interact to shape emotions.
- Vironment: liminal space between environment and invironment, bridging internal and external.
- Extended Reality (XR) unifies physical, virtual, and social worlds.
- Mersivity: technology connecting people, nature, and the physical world

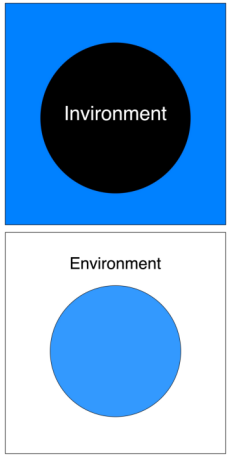


Figure 1: *Showcases concept of Invironment & Environment*

Approach

- The Muse S headband, equipped with EEG sensors, provides real-time tracking of brainwave activity, offering valuable insights into cognitive and emotional states.
- Facial emotions, a key component in understanding human affect, can be reliably detected using high-resolution webcams.

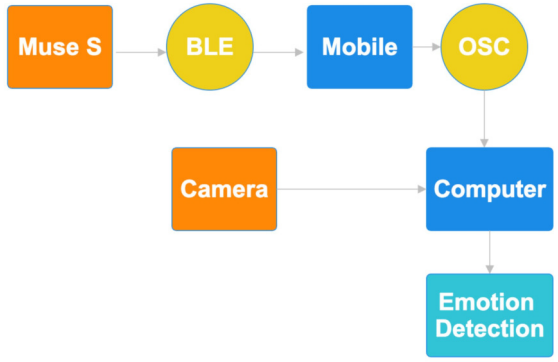


Figure 2: *Brain-wave emotion detection workflow*

- Emotion Recognition Using AI: Facial emotions are detected through computer vision techniques, leveraging convolutional neural networks (CNNs) to classify expressions like happiness, sadness, and anger with high accuracy and real-time performance.
- Multimodal Data Fusion: Integrating EEG signals from Muse S with facial emotion data enables a comprehensive analysis of internal (invironment) and external (environment) factors influencing emotional states

Figure 3: *Materials: (A): Muse S Headband (B): Webcam* (Creative Live Cam Sync 4K)

Results

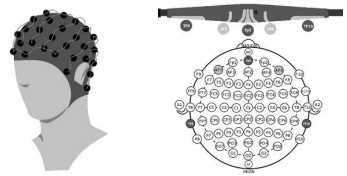


Figure 4: *Muse S Electrode Placements*

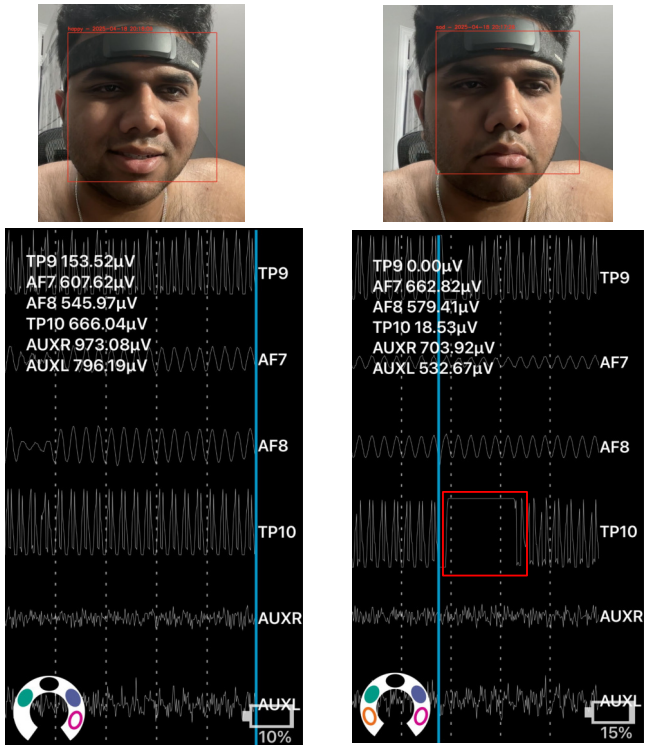


Figure 4: *Brain waves for Happy & Sad state with computer vision detecting the sentiment from facial expression*

References

[1] Mann, S. (2024). Advancing Technology for Humanity and Earth (+Water+Air). arXiv. https://arxiv.org/abs/2501.00074

[2]Mann, S. (2023). The eXtended Uni/Meta/Verse (XV) and the Liminal Boundary Between Environment and Invironment. PhilPapers. https://philpapers.org/rec/MANTEU-3

CONFERENCE CHAIRS

Steve Mann (PhD, MIT ’97, P. Eng., FIEEE), is widely regarded as The Father of the Wearable Computer” [IEEE ISSCC 2000]. He also invented wearable computing, as well as the hydraulophone as both an acoustic instrument and as water-human-computer interaction in his childhood in the 1960s and 1970s. In the 1980s he invented HDR (high dynamic range) imaging. In 1991 Mann and Charles Wyckoff invented, and coined the term, X-Reality (XR as eXtended Reality). Mann is a Fellow of the IEEE, a founding member of the IEEE Council on eXtended Intelligence (CXI), and a tenured full professor in the Department of Electrical and Computer Engineering at the University of Toronto. He is also the recipient of the IEEE Consumer Electronics Award (2025) as well as the Lifeboat Foundation Guardian Award.

Michael Condry (PhD, Yale ’80, LFIEEE), is a distinguished technologist and leader whose career spans both academia and industry. At AT&T Bell Laboratories he served as the chief architect for the BELLMAC 32 processor, one of the pioneering RISC designs of its era. He later joined Sun Microsystems and then Intel, where he rose to become Chief Technology Officer of the Client Division, the largest division of the company at the time. In academia, he has held teaching and research appointments at Princeton University and the University of Illinois at Urbana–Champaign. Condry has been deeply engaged with IEEE throughout his career, serving as a Senior AdCom board member and Past President of the IEEE Technology and Engineering Management Society (TEMS), as well as a member of the IEEE Industrial Electronics Society (IES).

PUBLICATION CHAIR

Nishant Kumar (B.A.Sc, UofT 2024), is a researcher and developer specializing in brain computer interfaces (BCIs), wearable health technologies, and signal processing. He has led and contributed to multiple projects involving EEG acquisition systems, edge computing, and adaptive environments driven by physiological signals. His work spans practical engineering implementations using devices like the Muse EEG headband and ESP32 microcontrollers, often focusing on scalable, open-access solutions. He is currently an M.A.Sc. student under the supervision of Dr. Steve Mann and an IEEE student member.