Adaptive Thermal Haptics:

Peltier-Driven Thermal Haptics for Virtual Reality

DAWSON L. HARRIS and HITENDER K. OSWAL, University of Washington

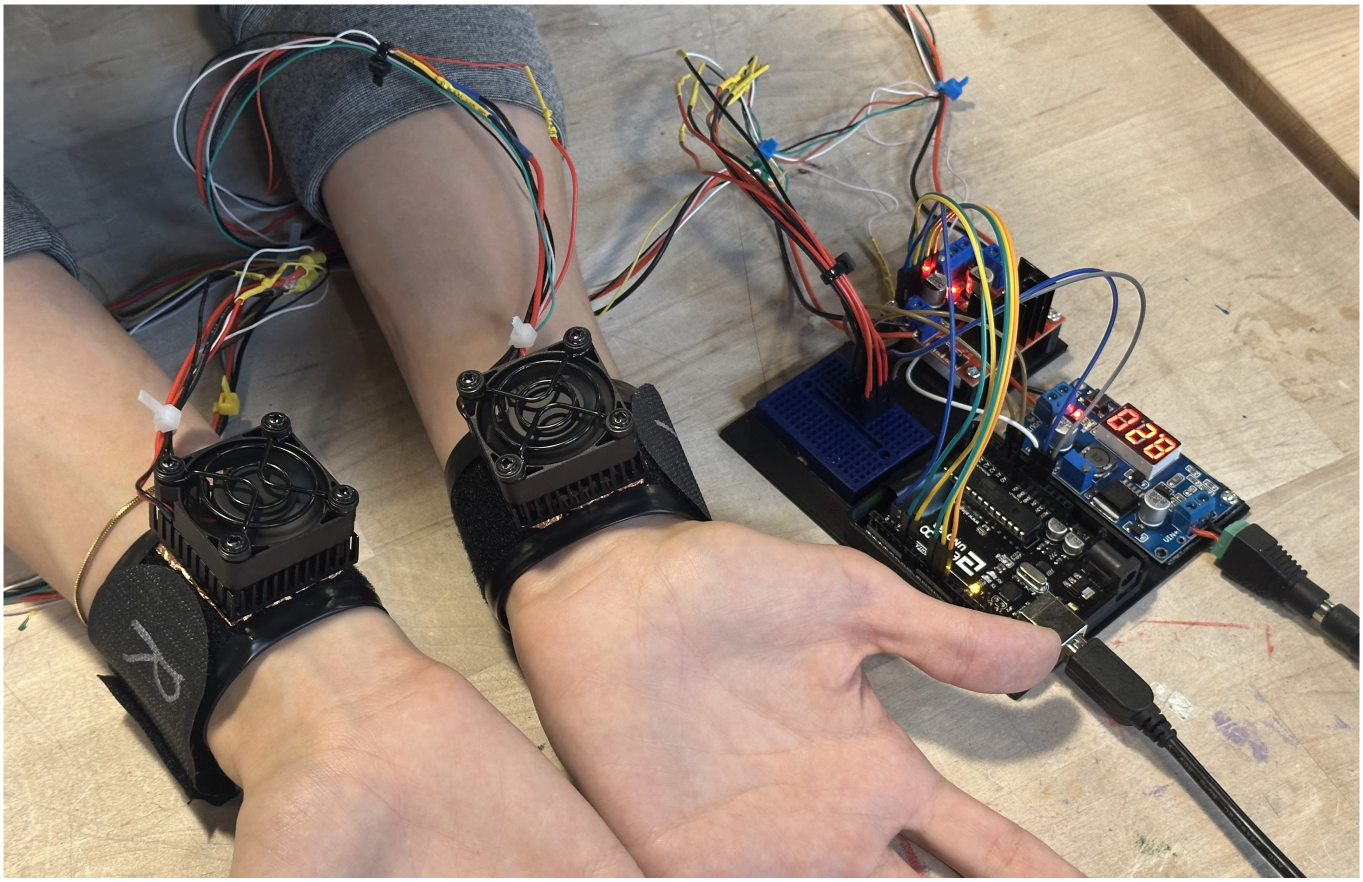

Fig. 1. The adaptive thermal wrist straps and electronics including Peltier cells, H-bridge, voltage regulator, power supply, temperature sensors, and Arduino.

Recent VR advances increasingly rely on multimodal feedback to enhance user immersion. This paper introduces adaptive thermal wrist straps that deliver realistic thermal haptics to the user's wrists, simulating temperature changes corresponding to virtual objects without impeding hand tracking or controllers. By combining a Peltier module, temperature sensors, and heat dissipation elements, each wrist strap dynamically adjusts its output under Arduino-based control, driven by real-time data from a Unity environment. Preliminary tests demonstrate an operating temperature range of 45 to 76 °F and real-time feedback. Though wall-powered, reducing mobility, offloading power and electronics minimizes on-body weight. Future work could improve power efficiency, widen temperature ranges, and use advanced control algorithms to provide more seamless thermal haptics and further enhance immersion.

Authors' address: Dawson L. Harris, dipson@cs.washington.edu; Hitender K. Oswal, hitender@cs.washington.edu, University of Washington.

1 INTRODUCTION

Virtual reality (VR) systems have primarily focused on providing users with realistic visual and auditory stimuli to create immersive, engaging environments through head-mounted displays. However, with the advancement of modern VR technology allowing for smaller and lighter devices—and the expanding interest in VR for different applications—there is a market for additional sensory modalities to further enhance immersion and improve the user experience. One of these modalities is temperature. Thermal cues have emerged as a particularly compelling avenue for exploration, as temperature feedback can more closely approximate real-world interactions and heighten the sense of presence within virtual environments, especially when interacting with objects and other stimuli.

Thermal cues are pivotal in how humans perceive and interpret their surroundings, influencing comfort, arousal, and emotional engagement. When interacting with objects in VR, the difference between feeling ambient temperature through a regular controller and actually perceiving the cold metal or warm fabric can drastically alter one's sense of realism as all materials have different thermal profiles. Realistic temperature feedback can also promote natural exploration habits and motor actions by increasing immersion.

Despite these benefits, implementing reliable, responsive, and comfortable thermal feedback systems remains an engineering and design challenge. Haptic gloves are the most common type of accessory to increase modalities. However, they are often prohibitively expensive, and thermal feedback requires significant space and power. To address these limitations, we created adaptive thermal wrist straps that offer accurate thermal cues for both hands without impeding access to other accessories, such as controllers or hand-tracking.

By simulating the temperatures a user would experience when touching or interacting with objects in virtual environments, the adaptive thermal wrist straps aim to increase immersion as a cheap addition without compromising other accessories or the performance of other hardware. This paper describes the design, implementation, and performance of these adaptive thermal wrist straps.

1.1 Contributions

- We created adaptive thermal wrist straps using Peltier cells for VR applications as well as the related electronic hardware to power, control, and regulate it.
- We created a temperature control program for the Arduino that maintains a desired temperature using the Peltier cells and temperature sensors.
- We integrated the temperature control system into Unity to control the wrist straps over a serial port. We also created a demo to showcase our project and its capabilities.

2 RELATED WORK

Peltier cells are the leading technology for simulating thermal cues for VR applications. There have been many different prototypes over the last decade attempting to solve this problem.

2.1 Commercially Available Devices

There are few commercially available devices that include thermal cues. One such device is the TouchDIVER haptic gloves which have haptic, texture, and thermal cues however they retail from $3,000 to $5,000 [weart 2024]. While these devices provide an excellent experience, their cost poses a serious barrier for most users. Individuals new to VR would benefit from a more realistic haptic experience at a reasonable price. As such, we focused on building a system that maximized price to performance.

2.2 Research Prototypes

Other non-commercial prototypes have also attempted to solve the problem of thermal cues in VR. Peiris, et al. integrated thermal haptic feedback to a head-mounted display for directional cues using 5 Peltier cells placed along the top and side of the headset [Peiris et al. 2017]. In this study, the Peltier cells were primarily for directional hints and feedback, rather than thermal cues themselves. Ranasinghe, et al. created a neck-mounted ambient temperature accessory for head-mounted displays as well as a wind accessory using fans [Ranasinghe et al. 2017]. Their main focus was on replicating the ambient temperature of the user's virtual environment, hence the location of the temperature accessory. Lee, et al. created a flexible and stretchable thermo-haptic device for use in haptic gloves [Lee et al. 2020]. Their device aimed to improve immersion by providing accurate thermal properties for objects in VR. While these prototypes tackle different challenges related to thermal cues, their focus is largely on using thermal cues for other purposes or on the temperature of the environment overall rather than providing thermal feedback to the user's hands. The work of Lee, et al. is most similar, with the primary distinction being that their focus was on thermal profiles of objects rather than the temperature of items and environments.

3 METHOD

Our project had many design considerations while building the hardware and software. We encountered various problems during the building process that we solved through different means as well as several safety concerns and safety mechanisms that we implemented to ensure safe operation. We will outline these challenges and how they were addressed.

3.1 Hardware

A major component of our design was affordability. As such, we aimed to minimize the components that we needed and to find easily accessible commercial products. This motivated us to choose items from vendors such as Adafruit and Amazon which can be accessed by anyone. Our total cost of parts was approximately $100, however, this cost includes items that were sold in sets greater than we needed, so the cost of the actual hardware in our prototype is closer to $75.

Our original prototype also placed the Peltier module in the palm of the user's hand in the form of a glove. However, this restricts the movement of the user, making it more uncomfortable to interact with controllers or use hand-tracking. Thus, our final design places the Peltier modules on a wrist strap placed on the inside of the user's wrists. This allows for complete freedom of the user's hands as well as an unobstructed view for hand-tracking algorithms to work.

One of the largest safety concerns we had was the Peltier module's max temperature differential of 65 °C. In order to avoid rapid temperature changes that could harm the user, we avoided running it at the maximum voltage of 5V. We instead lowered it to a safer 4V to 4.5V. This lowers how rapidly the Peltier can change temperature as well as the maximum differential that it reaches, reducing the chances of the user experiencing an unintended temperature.

3.2 Software

The software side of our project consists of Arduino and C# code for Unity. We will delve into the different algorithms and safety considerations that were used in each of these.

3.2.1 Arduino Code. Our Arduino code manages all logic related to the Peltier module and the safety of the user. For this reason, we needed to ensure that we could read an accurate temperature reading from our sensors. We used the following equation to convert the 10-bit Arduino reading (A) to temperature in Fahrenheit (°F):

$$^{\circ}F = \frac{9}{5}((100A * \frac{5}{1024}) - 50) + 32$$

The temperature reading is always paired with the desired temperature for safety.

Our original design used PWM to modulate the strength of the Peltier module, however, we learned that the Peltier module could be damaged from rapid on/off cycles. We opted to power the Peltier at max power to either cool or heat depending on the current temperature and the desired temperature. When the delta of the current temperature and desired temperature is less than 2 °F, the Peltier module is turned off. The temperature sensor is continuously monitored to maintain the desired temperature within 2 °F. In the case of rapid heating or cooling resulting in overshooting of the desired temperature, the opposite action is activated to reach equilibrium.

3.2.2 Unity Code. Our Unity code communicated to the Arduino to active the adaptive thermal wrist straps in sync with the VR environment. Since the Arduino is connected to the laptop over a USB port, there is no way for the Arduino to know if a Unity scene is running or not. Therefore, we implemented a pinging system that sends a message to the Arduino every second with an activation (or deactivation) signal. The adaptive thermal wrist straps can be toggled on or off using a keybind on the laptop. If the Arduino does not receive an activation signal for more than 2.5 seconds, it deactivates itself. This ensures that the adaptive thermal wrist straps deactivate in the case of Unity crashing or exiting.

Whenever an object is touched, the Unity code sends information containing the hand that touched the object and the temperature of the object that it touched. The Arduino is then able to activate the adaptive thermal wrist straps to provide thermal haptics. When the user stops touching the object, the adaptive thermal wrist straps are returned to ambient temperature.

The Arduino provides diagnostic data to Unity to monitor the state, temperature, and power of the adaptive thermal wrist straps.

4 IMPLEMENTATION DETAILS

Our project has two major components, the hardware of the adaptive thermal wrist straps and the interface with Unity for the demo.

4.1 Adaptive Thermal Wrist Straps

The adaptive thermal wrist straps are centered around 5V 1A Peltier modules from Adafruit. We ran them at 4.4V to prevent extreme temperature changes that could pose a danger to the user. Each Peltier module was paired with a TMP36 temperature sensor to continuously monitor the temperature that the user is exposed to and a 40mm heatsink and fan to dissipate heat. Copper tape is placed on the user side to prevent direct contact with the Peltier and spread the thermal cue to a larger area on the wrist. Both Peltier modules are powered by a single L298 H-Bridge. The H-Bridge is powered by an LM2596 adjustable voltage regulator set to lower 12V to 8V. The H-Bridge has a large amount of resistance, so this 8V outputted by the voltage regulator reaches 4.4V when it reaches the Peltier modules. The voltage regulator is powered through a barrel jack to screw terminal adapter and a personal 12V barrel jack power supply. Control of the H-Bridge and temperature sensors is managed by an Arduino Uno. The Arduino Uno receives commands from a serial connection to a laptop.

The adaptive thermal wrist straps have a software temperature limit of 35 to 76 °F however in real-world scenarios the cold thermal cue does not go below 45 °F due to the heat radiated from the user, as well as heat creep—term used for heat from the hot side of the Peltier overpowering the coldness of the cold side—from the opposite side of the Peltier. When the software temperature limit is exceeded, only the opposite action can be activated. This allows overshooting in excess of the maximum allowed temperature to be quickly corrected actively rather than passively. During our testing, we found 67 °F to be an accurate ambient temperature in most environments, however, this can be easily adjusted based on the temperature of your room.

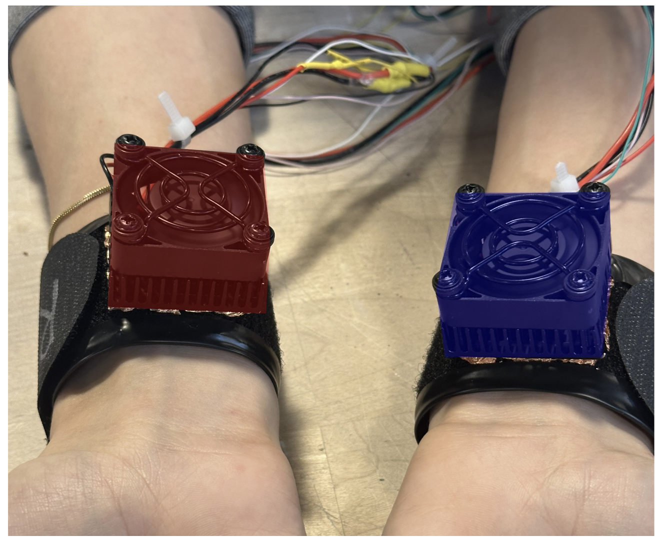

Fig. 2. The adaptive thermal wrist straps strapped onto a user's wrists.

4.2 Unity Demo

The interface between Unity and the adaptive thermal wrist straps was coded in C#. A demonstration was built in Unity using a Windows laptop in order to use SteamVR. Due to the need for a serial port connection to the Arduino Uno, the application needs to run on a computer and then streamed to the headset. The Ardity package was used to manage communication over the serial port to the Arduino Uno [Wilches 2024]. We used the Meta Quest Pro of one of the authors to deploy our environment.

5 EVALUATION OF RESULTS

The adaptive thermal wrist straps performed exceptionally well. Although we initially planned on running the Peltier modules at a low voltage such as 4V, we were able to raise it to 4.5V to achieve a faster temperature response time. The Peltier modules have a temperature range of 35 to 85 °F and achieved [#] °F/s during testing going from 40 to 80 °F. Feedback from preliminary user testing during development yielded positive feedback with comments on its rapid response and clear temperature differential.

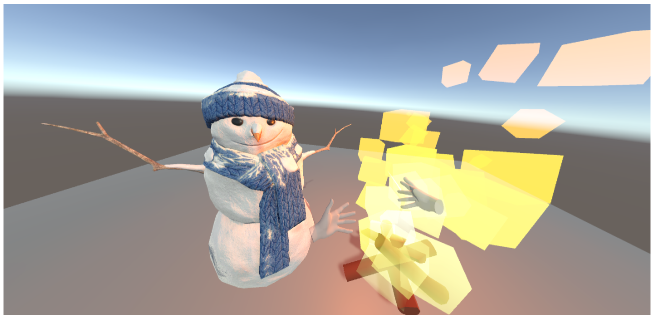

Fig. 3. The Unity demo with adaptive thermal feedback.

6 DISCUSSION OF BENEFITS AND LIMITATIONS

The biggest limitation of the adaptive thermal wrist straps is that they are wall-powered rather than having onboard batteries. This limits the user's movement to the length of the cables that connect to the wrist straps. However, this reduces the weight of the electronics worn by the user significantly as all power and control hardware is located off of the wrist straps. The added weight of batteries and electronics would pose a challenge for users as well as be limited by battery life. Being wall-powered allows it to run with more power, allowing for more effective thermal cues while remaining lightweight for the user. Another issue is the heat outputted by the H-Bridge and voltage regulator. Their heatsinks get rather hot and require good ventilation. Our design does not enclose them, allowing fresh air to reach them. However, a user-mounted approach would need to prevent users from touching these components while providing them with fresh air.

7 FUTURE WORK

Our hardware had a large amount of power wasted due to resistance in our components. Future iterations of such designs could utilize more efficient hardware, particularly H-Bridge and voltage regulator, in order to make battery-powered designs more feasible. This would also make the design suitable for commercial use as an accessory. There is also work to be done on increasing the intensity of the hot thermal cue while maintaining safety. Our design preferred safety, so the maximum temperature of 76 °F could be increased. Additionally, heat creep remains an issue, and there is work to be done on finding more effective cooling solutions for the outward-facing side of the Peltier.

Another possible improvement could be in temperature control. Since the Peltier does not perform well with PWM signals, the voltage could be modulated instead. This would improve temperature accuracy, smoothen the temperature curve, and reach a temperature equilibrium quicker as systems such as PID control could be implemented.

8 CONCLUSION

This work shows that adaptive thermal haptics can enhance the immersion of VR for all users affordably and unobtrusively. By integrating Peltier modules, temperature sensors, and a compact heat dissipation setup into wrist straps controlled via Arduino and Unity, users experience quick, localized temperature changes accurate to their VR environment. Though currently reliant on external power, these adaptive thermal wrist straps can be improved to be more efficient and portable. They show how multimodal experiences can be integrated into already existing hardware, offering a promising avenue for improving immersion in VR.

ACKNOWLEDGMENTS

We would like to extend a sincere thank you to Douglas Lanman and John Akers for their support, including providing resources and financial contributions that made this project possible.

REFERENCES

Jinwoo Lee, Heayoun Sul, Wonha Lee, Kyung Rok Pyun, Inho Ha, Dongkwan Kim, Hyojoon Park, Hyeonjin Eom, Yeosang Yoon, Jinwook Jung, Dongjun Lee, and Seung Hwan Ko. 2020. Stretchable Skin-Like Cooling/Heating Device for Reconstruction of Artificial Thermal Sensation in Virtual Reality. *Advanced Functional Materials* 30, 29 (2020), 1909171. https://doi.org/10.1002/adfm.201909171 arXiv:https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.201909171

Roshan Lalintha Peiris, Wei Peng, Zikun Chen, Liwei Chan, and Kouta Minamizawa. 2017. ThermoVR: Exploring Integrated Thermal Haptic Feedback with Head Mounted Displays. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 5452–5456. https://doi.org/10.1145/3025453.3025824

Nimesha Ranasinghe, Pravar Jain, Shienny Karwita, David Tolley, and Ellen Yi-Luen Do. 2017. Ambiotherm: Enhancing Sense of Presence in Virtual Reality by Simulating Real-World Environmental Conditions. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1731–1742. https://doi.org/10.1145/3025453.3025723

weart. 2024. TouchDIVER G1 haptic glove. https://weart.it/haptic-vr-products/touchdiver-haptic-glove/

Daniel Wilches. 2024. Ardity: Arduino + Unity communication made easy. https://github.com/dwilches/Ardity?tab=readme-ov-file