MultiKnob — A Knob for Multiplexing Rotation Inputs by Multitouch-based Grasp Recognition

Edgar Gellert
Technische Hochschule Köln
Gummersbach, NRW, Germany
edgar.gellert@th-koeln.de

David Petersen
Technische Hochschule Köln
Gummersbach, NRW, Germany
david.petersen@th-koeln.de

Matthias Böhmer
Technische Hochschule Köln
Gummersbach, NRW, Germany
matthias.boehmer@th-koeln.de

ABSTRACT

Rotary knobs are used on many devices. However, while we enrich the functionalities of smart devices, controlling such functionalities by using the rotation as sole input appears to be rather limited. Therefore, we created a physical rotary controller which enriches the input capabilities of a rotary knob, called *MultiKnob*. It takes the count of applied fingers as an input parameter for multiplexing rotation. The paper discusses three iterations of our prototyping: Beginning with an attempt to extend an existing rotary controller using conductive paint, to the development of several conductive 3D printed prototypes for use on a smartphone display, to a cylinder sensing multi-touch based on infrared light. We discuss possible applications and limitations of a knob which allows to recognize a user's grasp.

CCS CONCEPTS

• **Human-centered computing → Haptic devices**.

KEYWORDS

rotary knob, input device, rotation, multi-touch, grasp, usability, user experience

ACM Reference Format:
Edgar Gellert, David Petersen, and Matthias Böhmer. 2022. MultiKnob — A Knob for Multiplexing Rotation Inputs by Multitouch-based Grasp Recognition. In *Mensch und Computer 2022 (MuC '22), September 4–7, 2022, Darmstadt, Germany*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3543758.3547561

1 INTRODUCTION

Physical rotary knobs are used on many devices for control and interaction: e.g., on radiators, on radio equipment, on washing machines, for dimming light and in cars. However, input capabilities of rotary knobs are rather limited. Apart from tilting or pressing the knob, input is essentially limited to controlling one input variable continuously. However, devices today are enriched with extended functionalities and the control of such requires additional input capabilities; e.g., within a car we do not only have to set volume and frequency, but also zoom into maps, select contacts within an address book or switch between applications.

In this work, we set forth to develop an knob which allows to multiplex rotary input by recognizing a user's grasp while rotating the knob, called *MultiKnob*. We contribute the concept of *MultiKnob*, present three iterations of developing prototypes and discuss possible applications.

2 RELATED WORK

Various work has already been done in the context of physical rotary controllers with regard to shape [2, 7], performance [8], grasp [3, 5] and interaction posture [1].

van Oosterhout and Hoggan [7] studied the so-called DynaKnob, a shape-changing rotary controller. A performance study took place that evaluated the duration and accuracy of interaction with DynaKnob for visual and non-visual interaction. In addition, an analysis took place of how different combinations of knob shape and haptic stimulus affect the way users physically interact with the knob. Another work in the shape-changing field is by Kim et. al. [2]. They presented a shape-changing device that can change its shape between a rotary controller and a slider.

Kim et. al. [3] analyzed whether there is a discernible pattern in the positioning of the fingers (as per Napier's theory[5]), how subjects approach the device, how the angle of hand movement changes during interaction, and how many fingers were used during grasping and interaction.

Krieger et. al. [4] investigated in a study whether the number of fingers and the direction in which the fingers are grasped have an influence on human performance in haptic rotation. They showed that more fingers lead to better accuracy during interaction.

Voelker et. al. [8] investigated the performance between a virtual touch rotary controller and a physical rotary controller. They showed that physical rotary controllers are 20% superior to a virtual rotary controller in task processing speed. The superiority remains even when the subjects use the rotary controller blindly.

Gurari and Okamura [1] did not simply study a rotary controller, but the direct interaction with a device – the turning, the grasping, the arm movement, the speed, i.e. the basic strategy in turning a rotary controller. Their finding that a parallel arm angle and a smaller controller size lead to a more distal arm movement was also applied in our work.

Based on related work, we have determined that, for example, an appropriate diameter for our prototype is 6cm. A larger diameter could cause cramping of the wrist and fingers. Smaller diameters would force a smaller number of fingers during interaction.

The selection of papers mentioned here shows that physical knobs continue to be part of the research. However, all of these

MuC '22, September 4–7, 2022, Darmstadt, Germany

ACM ISBN 978-1-4503-9690-5/22/09…$15.00
https://doi.org/10.1145/3543758.3547561

systems limit their interaction to rotating and grasping. Our motivation is to develop a knob that not only can be turned, but also responds to the number of fingers that are applied.

3 CONCEPTUAL IDEA

When a rotary knob is grasped, it is touched by the user's fingers. Therefore we add the number of fingers used as another interaction dimension to a physical knob. In our research work, we mainly focused on the automotive context. The infotainment system of a car provides an excellent environment for using a rotary controller designed for finger recognition.

There are various ways to implement finger recognition as a single function or a general set of functions. For example, the various functions in the infotainment system can be mapped to the number of fingers applied. However, this would cover at most five different types of functions. To deepen the range of functions, the final system might take different contexts into account (these could also be applications in the infotainment system). For example, the user could switch between contexts/applications with five fingers. With four fingers the focused element could be changed. The remaining fingers could then trigger dedicated functions within the context or focused UI element.

The recognition of rotary gestures should also expand the range of functions. Thus, short commands can be integrated whose actual function call would actually require a longer interaction time. It would even be conceivable to allow users to train gestures into the system themselves. Finally, the rotary control should support use according to the controlled application by providing parameterizable tactile feedback.

Mapping the number of fingers to functions, gesture recognition, and also tactile feedback will need to be worked out in future work. In addition, further studies will determine the cognitive load during the interaction and how this can be reduced through supplemental feedback.

4 PROTOTYPE

4.1 First Prototype: Capacitive prototype

During the development phase, multiple iterations of the prototypes took place. For the first prototype we used an existing USB rotary knob. We tried to extend the knob's functionality by experimenting with built-in capacitive sensors of a microcontroller in combination with some copper foil and conductive ink. This prototype is shown in figure 1. As soon as the user is touching the material, the sensor will notice a change in capacity: It will change as a function of the occupied area. We aimed on identifying the number of touching fingers based on this value.

It has turned out, that this approach did not detect the number of touching fingers reliably, because of hardware problems. As an example, close proximity, but not touching can result in a detected finger. In short, we were not able to map electronic properties reliably to the correct number of fingers. Hence, we set forth to develop better capacitive sensors in the second iteration of the prototype.

4.2 Second Prototype: 3D Print Prototypes

The second prototype consists of a 3D print in combination with a smartphone screen. The principle behind this was to make a 3D print with some conductive material so that it could be used on a smartphone display. As soon as a user touches the conductive material, the touch is forwarded and detected by the screen. Delegating the touch to a smartphone has the advantage to be able to use one of the most developed capacitive sensors on the market.

4.2.1 Conductive Materials: Foil, Wire, Ink, Filament. In the first iterations, aluminum foil, wire and conductive ink was used to transmit the touch to the display. These conductive materials were embedded in 3D printed bodies, that are not conductive themselves.

Unfortunately, the conductive materials could not be applied as precise as needed, so that the 3D print does not always lay completely flat on the screen. As a result, occasionally some touches went undetected.

To address this problem 3D-printable conductive material was used in the subsequent iterations. The conductive components, which were previously created with aluminum foil, wire and conductive ink, were now 3D printed as separate conductive elements. These were inserted into the non-conductive knob-shaped body.

4.2.2 3D-printable conductive Filament. The conductive components were printed as L-shaped pins so that they can be guided along the outer edge to the underside of the object, like in the first iteration using aluminum foil in figure 2. It turned out, that iterations using the L-shape were error-prone. Thus the final iteration does not use the L-shape, instead the touch is forwarded to the screen by a pin-shaped element, as shown in figure 3.

4.2.3 Android Application. The counterpart to the 3D printed objects is an Android application running on a smartphone. The assembled 3D print is placed upside down on the device, so that the conductive ink or filament is touching the screen: As soon as the user touches the conductive material, the touch is forwarded to the screen. This results in touch events fired within the application. These events are processed to be able to detect the number of fingers currently touching the 3D print.

Additionally, the application tracks the position for one single touch point. This enables us to compute the rotation value, as soon as the user rotates the 3D print while holding it with an arbitrary number of fingers.

4.2.4 Preliminary Study. A small preliminary study was performed to measure the accuracy of the 3D printed prototype. The prototype movement was constrained so that it was only able to rotate around its own axis. Additionally subjects were not able to touch the devices screen directly. Thus all produced touch events were associated to the prototype.

It has been found that the prototype performed weakly for different sizes of hands. Some subjects touched multiple conductive components with one finger, resulting in detecting too many fingers. Other subjects touched between the conductive components (the white, non-conductive material), resulting in detecting too few fingers.

This problem was intensified when subjects rotated the prototype: The fingers might somewhat move around the controller,

Figure 1: First Prototype: Using conductive ink on an existing USB knob.

Figure 2: Second Prototype: First iteration using aluminum foil.

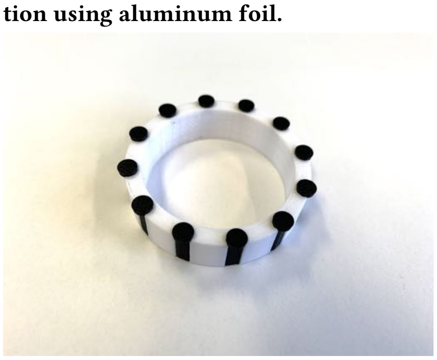

Figure 3: Second Prototype: Final iteration using conductive filament.

resulting in not touching a previously touched conductive component. This, again, results in not detecting the correct number of fingers.

4.3 Third Prototype: FTIR Prototype

The third prototype should be able to detect the finger no matter how big or small, or where exactly they are touching. To achieve that, the next prototype is based on the principle of Frustrated Total Internal Reflection (FTIR).

4.3.1 FTIR. Schoening et. al. [6] show in their paper a summary of various optical multi-touch surfaces for tabletop displays. The basic principle of FTIR is that infrared light is shone into an acrylic body. The infrared light is reflected inside the acrylic body and illuminates it. When a finger (or several fingers) touches the outer surface of the acrylic body, its refractive index changes, the infrared light exits the acrylic body and is reflected by the fingertip. The infrared light reflected from the fingertip can then be perceived by an infrared camera. Using a computer vision application based on OpenCV, the reflection of the infrared light can be subtracted from the background environment.

4.3.2 3D printed Setup. First considerations to transfer the FTIR principle into a rotatable controller are shown in figure 4. The concept of an FTIR rotating controller includes the acrylic body in the shape of a cylinder, a ring with 18 infrared LEDs, an optional infrared filter, an infrared camera with a fish-eye lens, and all this wrapped within a multi-component 3D print. Figure 5 shows the first prototype, the final setup is shown in figure 6.

The housing consists of several pluggable 3D printed components. This enabled faster prototyping as single components could be reiterated independent of each other and minimized the use of support material during the print.

4.3.3 Implementation. Figure 7 shows the unprocessed image, the camera is capturing. An OpenCV implementation processes this image and returns the resulting image show in figure 9 as well as some metadata like number of fingers and rotation in degrees.

For each frame the implementation runs through numerous steps. Firstly it greyscales the raw image and searches for two circles. A circle within the lid is used to determine the rotation from frame n to frame n+1. The other circle is the lid itself. It is used to limit the touchable area beneath it, represented in figure 9 with two white circles. This area is masked, so that further processing ignores all pixels outside that area. Within that area all pixels have a value between 0 (= black) and 255 (= white). All pixel values below 255 are set to 0 so that only pure white pixels remain, as shown in figure 8. On this picture a blob detection is applied. The blob detection is tweaked to find fingertips of different sizes and shapes. Finally the found blobs are highlighted. The number of blobs correspond to the number of fingers used to operate the controller.

5 DISCUSSION

The FTIR prototype allows us to recognize fingers on the outer surface of a round object using technology that was previously used on large surfaces. Various contexts can be described as suitable use cases for detecting and using the number of fingers on a physical rotary controller. As mentioned in section 3, mainly the automotive context was the focus of the project. However, other contexts are also conceivable. For example, *MultiKnob* could be used for video editing applications. Depending on the number of fingers applied, different functions are triggered during editing or even executed as a chain function one after the other if the task is executed with a different number of fingers. In this way, functions can be executed that build on each other without having to remove the hand from the rotary controller. Specifically, it could look like this: The editor can use five fingers to switch between tracks, four fingers to apply the cut, and finally three fingers to move the cut video element to its desired location.

Another example is CAD tools to design complex 3D objects. In addition to the traditional movements of the object in virtual space,

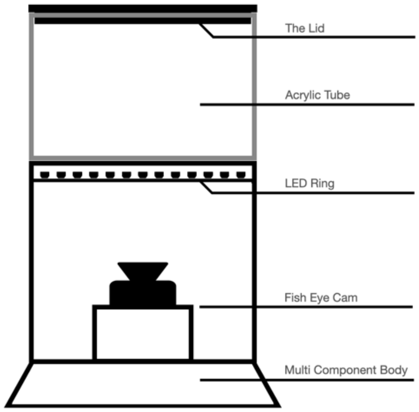

Figure 4: Concept of FTIR rotary controller.

Figure 5: Early iteration of the prototype.

Figure 6: Final assembled construct.

chained functions are also conceivable in this context. By stringing together rotations with different numbers of fingers, different functions can be applied to the designed body. For example, after

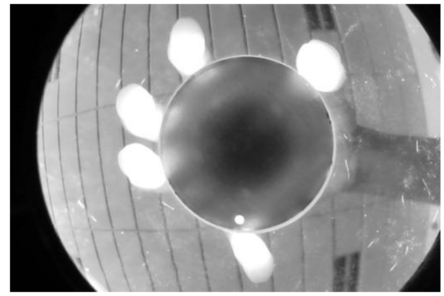

Figure 7: Raw camera image.

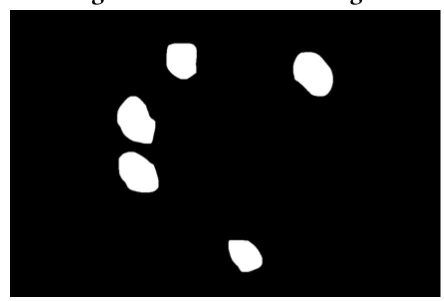

Figure 8: Extract white pixels only.

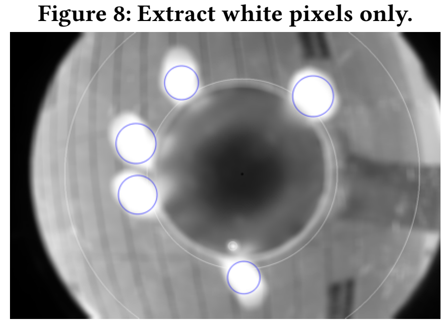

Figure 9: Blob detection.

selecting a body, a padding can be initiated and then its intensity can be varied with a different number of fingers.

Such a rotary controller can also be used as an advanced interface for the smart home. With five fingers, one could switch through the individual devices. When controlling a light bulb, it could not only be dimmed by turning the knob, but by varying the fingers the light could be manipulated in further properties, such as the color spectrum. There are a lot of use cases that can be enriched with a rotary controller and the recognition of the applied fingers.

5.1 Limitations

Since the system is based on infrared light, the more ambient light there is, the more susceptible it is to interference. In a darkened room, without direct sunlight and closed blinds (i.e., only indirect light sources), finger recognition is quite remarkable. However, this changes as soon as the system is used outdoors and the environment is filled with infrared light, or when too much light floods a room, be it direct sunlight or even strong ceiling lights. If there is too much ambient light, the FTIR-based system cannot recognize fingers reliable.

Furthermore, the processing by OpenCV is computational demanding. Single frames in the pipeline are processed sequentially and thus single-threaded. This can result in noticeable delays in interaction when running on a slow CPU. We are using a 2021 MacBook Pro with a M1 Pro SoC which processes the frames fast enough to prevent a noticeable delay.

6 CONCLUSION

In this work, we explored some options for finger recognition on rotary controllers. In the course of the process, several prototypes were created. The first prototype is not able to determine the correct number of fingers reliably. The second prototype is able to determine the correct number of fingers and rotation, but only if the user touches the conductive material in a prescribed way. In a short evaluation we found that this is not always the case. Thus a third prototype was developed, based on FTIR technology. This prototype shows the best finger recognition in contrast to the other prototypes.

We will continue to develop and optimize this FTIR based prototype and use it in future research. We aim to develop and evaluate an interaction design that can lead to new interaction patterns based on the number of fingers on a rotary controller.

ACKNOWLEDGMENTS

This work has been partly funded by the Central Innovation Programme for small and medium-sized enterprises of the German Federal Ministry for Economic Affairs and Climate Action under grant KK5028801LF0.

REFERENCES

[1] N. Gurari and A. M. Okamura. 2007. Human Performance in a Knob-Turning Task. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*. IEEE, 96–101. https://doi.org/10.1109/WHC.2007.71

[2] Hyunyoung Kim, Céline Coutrix, and Anne Roudaut. 2018. KnobSlider: design of a shape-changing UI for parameter control. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173913

[3] Hyunyoung Kim, Patrícia Deud Guimarães, Céline Coutrix, and Anne Roudaut. 2019. ExpanDial: designing a shape-changing dial. In *Proceedings of the 2019 on Designing Interactive Systems Conference*. Association for Computing Machinery, New York, NY, USA, 949–961. https://doi.org/10.1145/3322276.3322283

[4] Kathrin Krieger, Alexandra Moringen, and Helge Ritter. 2019. Number of fingers and grasping orientation influence human performance during haptic rotation. In *2019 IEEE World Haptics Conference (WHC)*. IEEE, IEEE, 79–84. https://doi.org/10.1109/WHC.2019.8816096

[5] John R Napier. 1956. The prehensile movements of the human hand. *The Journal of bone and joint surgery. British volume* 38, 4 (1956), 902–913. https://doi.org/10.1302/0301-620X.38B4.902

[6] Johannes Schöning, Peter Brandl, Florian Daiber, Florian Echtler, Otmar Hilliges, Jonathan Hook, Markus Löchtefeld, Nima Motamedi, Laurence Muller, Patrick Olivier, et al. 2008. Multi-touch surfaces: A technical guide. *Johannes Schöning, Institute for Geoinformatics University of Münster, Technical Report TUM-I0833* (2008).

[7] Anke van Oosterhout and Eve Hoggan. 2020. Reshaping Interaction with Rotary Knobs: Combining Form, Feel and Function. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*. Association for Computing Machinery, New York, NY, USA, 1973–1982. https://doi.org/10.1145/3357236.3395536

[8] Simon Voelker, Kjell Ivar Øvergård, Chat Wacharamanotham, and Jan Borchers. 2015. Knobology revisited: A comparison of user performance between tangible and virtual rotary knobs. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces*. Association for Computing Machinery, New York, NY, USA, 35–38. https://doi.org/10.1145/2817721.2817725