

A Short Guide to Modulated Light

Florian Echtler, Tobias Sielhorst, Manuel Huber, Gudrun Klinker

Technische Universität München
Institut für Informatik
Boltzmannstr. 3
D-85747 Garching, Germany

{echtler,huberma,klinker}@in.tum.de, sielhorst@googlemail.com

ABSTRACT

Many types of tangible interaction systems, such as interactive surfaces and gesture-based interfaces, are based on various kinds of optical tracking, using infrared illuminators and cameras. One drawback of these setups is that they suffer from problems common to optical trackers, such as sensitivity to stray environment light from artificial and natural sources. In this paper, we present a method to significantly enhance tracking robustness for those systems which employ active illumination. Through addition of a small electronic circuit which modulates the LEDs used to illuminate the scene, contrast can be significantly increased.

Author Keywords

input device, hardware, optical tracking, infrared

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User Interfaces—*Input Devices*

INTRODUCTION AND RELATED WORK

A significant amount of tangible interfaces is based on optical sensing, using either commercially available sensor systems or a camera whose images are processed with computer vision algorithms. In order not to disturb the simultaneous display of data visible to the user, these systems usually employ infrared light sources and cameras, thereby remaining invisible to the viewer.

As these setups can easily be built from off-the-shelf components in most cases, many researchers now have access to these novel input devices. Most setups require little more hardware than a CCD camera, a suitable filter and infrared LEDs. Well-known examples from the area of interactive surfaces include the work of Wilson [11], the Microsoft Surface [7] or the seminal paper by Han [4]. Other systems,

which do not focus on touch detection, but rather on tangible object detection are, for example, the work by Echtler et al. [3], reacTIVision by Kaltenbrunner et al. [6] or the MIT metaDESK [5].

Although widely used, these setups are mostly still susceptible to problems common to all optical trackers. One big drawback is that they are usually sensitive to environment light from sources like ceiling lamps or daylight. In this paper, we present a method to enhance the contrast between stray environment light and the light from active illumination, thereby effectively increasing the signal-to-noise ratio.

While our solution depends on adding a small electronic circuit to the setup, we believe that the construction is easy enough to pose no significant problems.

Surprisingly, the concepts presented here seem to be common knowledge in the computer vision community, especially in the industrially-oriented branch. Examples for commercial products which likely use this method include the ARTrack system [1] or the Optitrack cameras [8]. However, we have been unable to find concrete references or guides to these concepts. We therefore present them here, also to bring them to the attention of researchers and practitioners from the user interface community.

PULSED ILLUMINATION

The first requirement of the method presented here is that the LEDs which comprise the infrared illuminator are not operated continuously anymore. Instead, they are supplied with short pulses of high current. In fact, the pulse current is high enough that under sustained operation, the LEDs would be likely to suffer permanent damage after a few seconds.

Usually, these pulses have a duration of some hundred microseconds to few milliseconds. The reason for this mode of operation is that the high current throughput which is possible during the short pulse results in a much higher light output. An increase of one order of magnitude can easily be achieved.

The pulse duration and the following cooldown period should be kept as close to the specification as possible to prevent overheating of the LEDs. As modern computers usually are not equipped with the hardware nor software to handle such realtime tasks, we suggest using a 555 timer or a sim-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

TEI 2009, February 16 - 18, 2009, Cambridge, UK.

Copyright 2009 ACM 978-1-60558-246-7/07/0004...\$5.00.

ple microcontroller to generate the pulses. The 555 timer is an integrated circuit which generates electrical pulses of defined length and has been in use for decades. It has the advantage of requiring no programming prior to use, but is somewhat inflexible and error-prone, as the pulse duration is determined by two external components (resistor and capacitor) which often have considerable production tolerances. If a programming device is available, a microcontroller like the ATmega8 might be a better choice. It can be programmed in plain C code and offers the potential for additional features, such as an USB link that could be used to control the LEDs individually.

While the microcontroller is responsible for generating the correct pulse timing, a second-level switching element is necessary to propagate them to the LEDs, as none of the integrated circuits mentioned previously are capable of handling the high currents which momentarily occur. FETs (field-effect transistors) are particularly easy to integrate with logic circuits., We thus suggest using these as second-level switches. One example would be the IRF512 logic-level FET.

Finally, as an additional precaution against LED damage, a fuse with a *lower* rating than the expected pulse current can be inserted in series with the LEDs. Should the pulse generation fail for any reason, thereby continuously powering the LEDs, the fuse will blow immediately, thus protecting the other parts of the circuit. Although more current will flow through the fuse than it is rated for, it will not blow during pulsed operation, as the single pulses are too short. For convenience, it is advisable not to use a common fusible, but rather a so-called polyfuse, which will reset itself after the power has been turned off.

In Figure 1, we present an example circuit diagram which uses the concepts mentioned above. It consists of a total of 5 components, excluding LEDs and power supply, and can easily be built on a breadboard. The two supply voltages of 5 and 12 Volts can be drawn from a standard PC power supply. For details on how to calculate the capacitor/resistor values, see [10].

The 555 timer may not be the best solution in some cases. Yet, an introduction to microcontrollers is beyond the scope of this paper. We have therefore focused on this basic, but widely usable variant.

CAMERA SYNCHRONIZATION

Although pulsing the LEDs already increases total light output, this in itself is not yet sufficient to gain a significant contrast boost relative to stray environment light. To achieve this goal, the pulses also should be synchronized with the camera in such a way that exactly one pulse is emitted for each camera frame and that its duration is equivalent to the camera's exposure time.

As the LEDs are usually brighter by about one order of magnitude during the pulse, the contrast ratio with respect to environment light is also significantly higher. If the camera exposure time is longer, stray environment light is accumulated

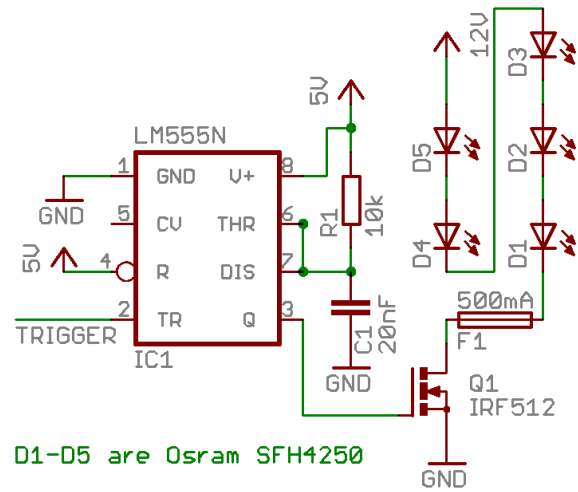


Figure 1. Pulse Generation Circuit

during the cooldown pauses until the next pulse, thereby decreasing the contrast ratio again. In Figure 2, example light curves for comparison of the three modes are shown.

In these examples, we define the contrast ratio as the quotient $c = \frac{I_i}{I_a}$, with $I_i = \int_0^t f_i$ being the *illuminated* intensity of the objects which we want to capture and $I_a = \int_0^t f_a$ being the *ambient* intensity caused by lamps, daylight and other external sources. t is the camera exposure duration for a single frame. In all cases, we assume a constant ambient light function, $f_a = c_a$, and therefore a constant ambient intensity $I_a = C_a$.

In the first mode, the LEDs are powered continuously, therefore f_i is also constant. To prevent overheating, the current flowing through the LEDs and consequently also the illuminated intensity may not exceed a certain threshold.

In the second mode, the LEDs are pulsed rapidly with high current and cooldown pauses in between. Several pulses fall into one camera frame. While the light output is much higher than in the first mode, the cooldown pauses allow a significant amount of ambient light to be collected in between the pulses.

The third mode alleviates this problem by shortening the camera exposure duration to be approximately equal to the LED pulse duration. While this reduces the total captured intensity, the contrast is increased, as the light output during the pulse can be almost one order of magnitude higher than during continuous operation.

Of course, the camera needs a configurable trigger output and exposure duration for this method to work. This is not a problem with industrial cameras, which are usually based on the Firewire protocol and contain these as standard features. Depending on the camera model, it may even be possible to generate the entire control pulse with the trigger out-

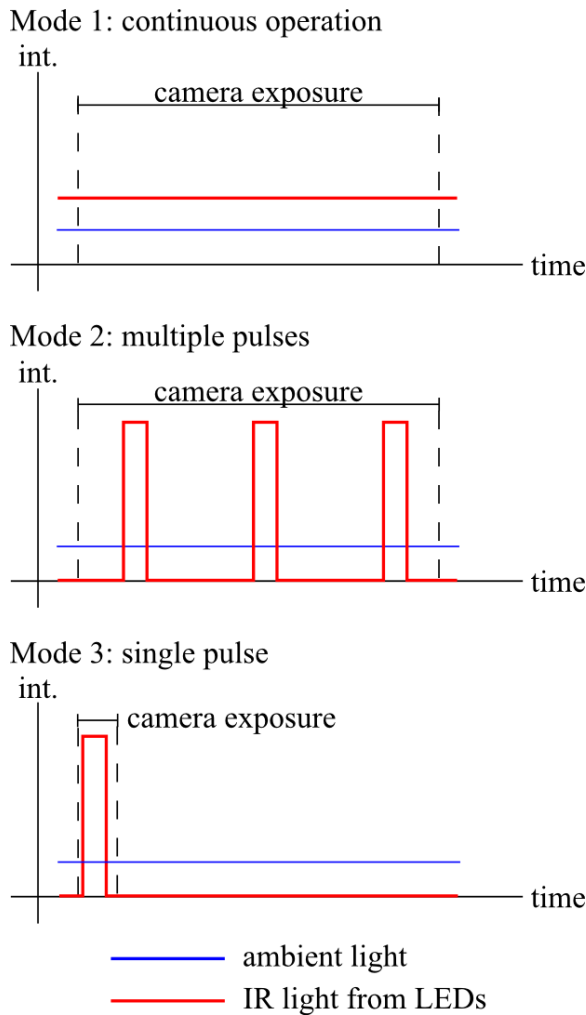


Figure 2. Light intensity curves for all three operation modes (not to scale).

put, thereby reducing the external circuitry to 2 components (FET and fuse).

In many cases, however, it would be desirable to use a simple webcam with such a setup due to price and availability. While most webcams can easily be converted to detect infrared light by removing the corresponding filter, a trigger output is not available. We have evaluated the possibility to add such an output to a Logitech Quickcam 5000. After opening the camera casing, we used a digital sampling oscilloscope (DSO) to check the signal lines on the internal circuit board for output of periodic signals consistent with the camera frame rate. Although we were able to locate such an output signal and use it to drive a circuit like the one described above, the results were unsatisfactory. The reason is that webcams usually use a so-called rolling shutter, i.e. the image pixels are read line-by-line from the sensor, thereby spreading the capture of a single frame over a significantly larger amount of time. In contrast, industrial cameras use a global shutter, thereby capturing the entire frame in one shot. This is an unavoidable requirement for the presented

synchronization method. As a consequence, this unfortunately means that use of an industrial camera is currently necessary.

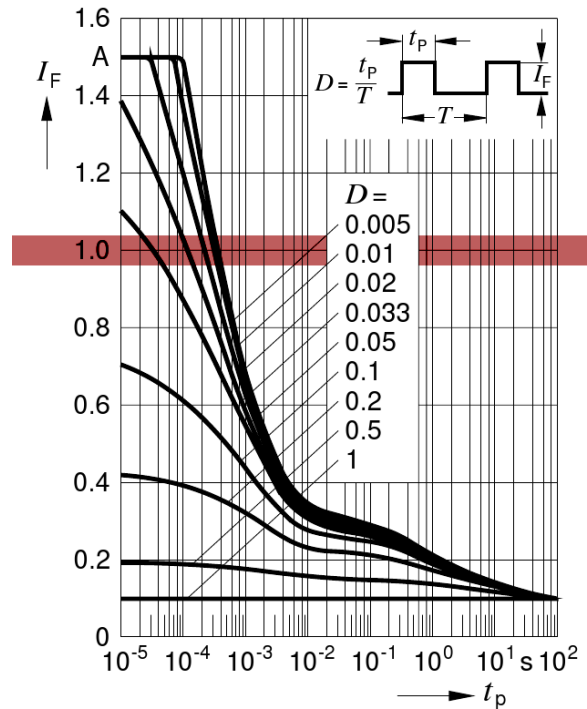


Figure 3. LED pulse capacity diagram

DETERMINING THE PULSE DURATION

At the example of Figure 3, we will now describe how to calculate the correct pulse/exposure duration for a specific camera and LED combination (Pointgrey Firefly MV and Osram SFH4250 LEDs). For a detailed view of the curves used, see [9].

Assume a frame rate of $f = 60\text{Hz}$. Therefore, one full pulse/cooldown cycle must have a duration of $D_{max} = \frac{1}{f} = 16.67\text{ms}$. We are operating the LEDs at a voltage of 2.4 V (12 V divided by 5 LEDs), resulting in a current of 1 A (the voltage-current ratio is also given in the datasheet). We now have to calculate the total cycle duration, based on the duty cycle for each curve and the allowed pulse duration at a current of 1 A. For example, at a duty cycle of 3.3%, the pulse duration is approximately $t_p = 120\mu\text{s}$ for a total cycle duration of $D = 3.6\text{ms}$. At a ratio of 1% with a pulse duration of $t_p = 250\mu\text{s}$, the total duration already rises to $D = 25\text{ms} > D_{max}$, which is too long. We therefore select a duty cycle of 2%, resulting in a pulse duration of $t_p = 200\mu\text{s}$ with a total duration of $D = 10\text{ms}$, which still offers a comfortable safety margin. Of course, the camera must be able to provide such short exposure times. This is the case with our model.

RESULTS

For a comparison of the continuous and single-pulse operation modes, see Figure 4. A single LED is viewed head-on

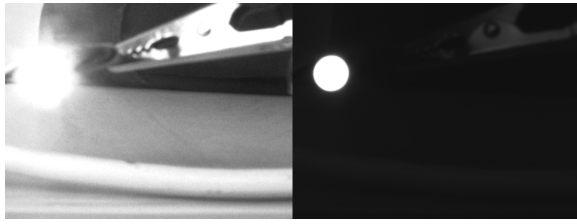


Figure 4. Images from continuous (left) and pulsed (right) operation.

by the camera. In both images, the LED is captured by the sensor with maximum brightness (255 in 8-bit mode). In the continuous mode, however, the brightness of the background is about 160, whereas in the pulsed mode, the background values are about 20, an eight-fold difference. Although the exact contrast ratio can not be determined from this example, as the illuminated pixels are saturated in both cases, we can safely conclude that the contrast can be increased at least by a factor of eight.

We have implemented this method in a real-world setup in an FTIR-based interactive table [2]. Previously, the system was unusable in anything except near darkness. Even the infrared component in normal daylight was sufficient to drop the contrast below usable levels. With the circuit and camera synchronization in place, we are now able to use the system while the ceiling lamps are turned on or the window blinds are open, even under bright daylight conditions.

CONCLUSION

In this paper, we have evaluated an easy-to-build hardware modification to increase robustness with respect to ambient light for infrared-based optical sensing. While this method has already been implemented in some commercial products, we believe that this guide offers valuable information to non-commercial users, especially in the research community. If a camera with trigger output is available, a simple circuit consisting of only five parts is sufficient to significantly increase contrast to environment light.

REFERENCES

1. Advanced Realtime Tracking. ARTrack. <http://www.ar-tracking.de/>.
2. F. Echtler, M. Huber, and G. Klinker. Shadow tracking on multi-touch tables. In *AVI '08: Proceedings of the working conference on Advanced Visual Interfaces*, pages 388–391, 2008.
3. F. Echtler and G. Klinker. Tracking mobile phones on interactive tabletops. In *MEIS '08: Proceedings of the Workshop on Mobile and Embedded Interactive Systems*, 2008.
4. J. Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*, pages 115–118, 2005.
5. H. Ishii and B. Ullmer. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *CHI '97:*

Proceedings of the Conference on Human Factors in Computing Systems, pages 234–241, 1997.

6. M. Kaltenbrunner and R. Bencina. reactIVision: a computer-vision framework for table-based tangible interaction. In *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 69–74, 2007.
7. Microsoft. Surface. <http://www.microsoft.com/surface/>.
8. NaturalPoint. OptiTrack. <http://www.naturalpoint.com/optitrack/>.
9. OSRAM Opto Semiconductors. SFH 4250 Datasheet. <http://www.osram-os.com/>.
10. T. van Roon. 555 timer tutorial. <http://www.uoguelph.ca/~antoon/gadgets/555/555.html>.
11. A. Wilson. PlayAnywhere: a compact interactive tabletop projection-vision system. In *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*, pages 83–92, 2005.