Time domain continuous imaging using LED sensels

Henry Dietz and Paul Eberhart and Alyssa Oretel
Department of Electrical and Computer Engineering, University of Kentucky; Lexington, Kentucky

Abstract

Time domain continuous imaging (TDCI) centers on the capture and representation of time-varying image data not as a series of frames, but as a compressed continuous waveform per pixel. A high-dynamic-range (HDR) image can be computationally synthesized from TDCI data to represent any virtual exposure interval covered by the waveforms, thus allowing both exposure start time and shutter speed to be selected arbitrarily after capture, which also enables extraction of video with arbitrary frame rate and shutter angle. Unfortunately, conventional sensors cannot directly implement TDCI capture, so earlier work focused on postprocessing conventional sensor output to approximate TDCI streams.

The current work describes the first direct implementation of TDCI sensing. The sensors discussed here are not image sensor chips, but prototype equivalent circuitry and control logic as low pixel count board-level sensor modules constructed using commodity components. A LED is used to implement each sensel, and each is sampled asynchronously independent of all other sensels by reverse biasing the LED to charge its inherent capacitance and then timing how long the photocurrent takes to reach a fixed threshold voltage. These open source LED-based TDCI sensor modules are used to construct stand-alone TDCI cameras, allowing performance measurements, tweaking of the control logic, and empirically verifying that true TDCI sensing is practical.

Introduction

In both film and digital photography, even when capturing the changes in scene appearance over time, the goal has generally been to capture an image or a sequence of images. Each image ideally represents the average appearance of the scene over a given exposure interval. Some shuttering methods only approximate that; for example, focal plane shutters and rolling electronic shutters produce slightly skewed timing across the frame. The goal remains sampling all points in the scene for the same time interval. In fact, the word "image" itself normally refers to *a representation or likeness fixed in time*. Time domain continuous imaging (TDCI)[1] is different.

What is TDCI?

Conceptually, the appearance of a scene is a material property that is sampled using photons. The goal in capture is not to count photons for a specific time period, but to use the photons to determine the appearance of each portion of the scene and to determine when each portion changes appearance. In other words, TDCI is about capturing a waveform per pixel describing how the corresponding portion of the scene changes appearance over time. These waveforms are generally highly compressible, because only an unpredicted change in the expected value of a pixel produces new data. Think in terms of each sensel, independent of all other sensels, sampling enough photons to determine when the corresponding pixel should change appearance. This is the type of sensor TDCI needs: one that directly produces an IMEV (IMage EVolution) data structure.

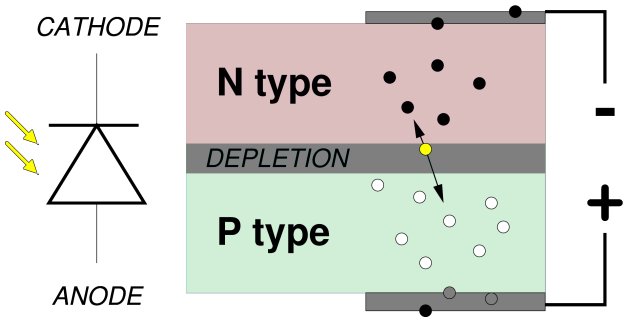

Figure 1. *PN junction produces hole+electron per photon absorbed*

Given an IMEV, TDCI algorithms can do a variety of novel and useful things. Given that there is a pixel value error model being applied to distinguish noise from scene content changes, a temporal sequence of samples with the same content can be averaged to produce a better estimate of the true pixel value. This tends to increase dynamic range as well as decreasing noise. It is also possible to render an image integrating over any time period covered by the IMEV data, selecting the integration interval *after* capture. This allows rendering videos with arbitrary frame rates and shutter angles, even angles greater than 360°.

Sensor Alternatives

When TDCI was first proposed in 2014[2], several potential sensor implementation methods were discussed including the ideal goal of creation of a large-format sensor using a segmented solar cell with a nanocontroller under each sensel. Each solar cell sensel is essentially a large-surface-area PN junction that has a photovoltaic response creating a hole+electron pair as each photon is absorbed, as diagrammed in Figure 1. The electron is drawn into the N region and the hole to the P region, with a current thus flowing from the N contact to the P contact, where the electron and hole recombine. Alternatively, the PN junction can be used as a photoconductive sensel in which light increases leakage for the reverse-biased junction and the time to reach a threshold can be measured digitally. It was this photoconductive mode that we wanted to employ, compressing the output by

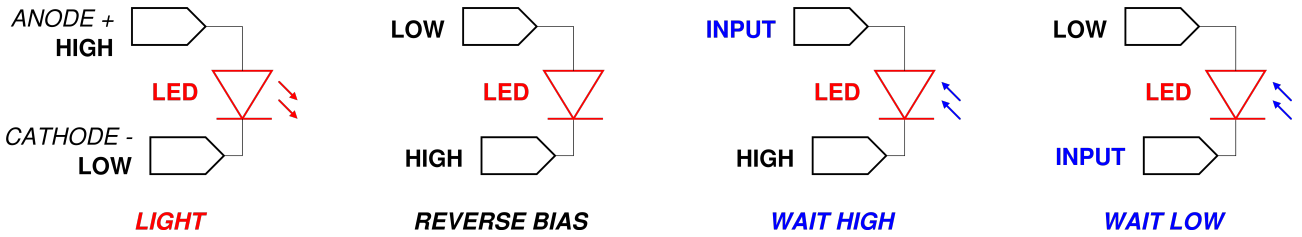

Figure 2. *LEDs don't just emit light; they also support multiple modes of operation as sensors*

only emitting a record when a sensel's expected value changed. Unfortunately, work toward building a 4x5" 500MP segmented solar-cell sensor chip was shelved when changes in the operation of the University of Kentucky Center for Nanoscale Science and Engineering made it impractical to fab the device there.

Since then, TDCI processing has been explored and advanced in a variety of research publications[3, 4, 1, 5, 6, 7, 8]. However, all these later works were applied as postprocessing of time sequences of conventionally-captured images. Thus, the search for a simple, natively IMEV, TDCI sensor has continued.

There are many near misses for the type of sensor desired. A 1994 paper[9] describes photoconductive measurement of a photodiode sensel implemented in a CMOS imager. However, the details of that system do not result in an effective IMEV data stream. Instead, the pulse frequency is determined by a set of per-pixel counters operating within a fixed exposure interval, essentially producing conventional images as output. Time-to-first spike (TTFS) CMOS imagers[10] are very similar, but use the time to the first pulse within a fixed exposure window to obtain the value for each pixel within a captured image. More recent work has basically moved the threshold down to a single photon arriving, such as the jot-based QIS (Quanta Image Sensor)[11]. Unfortunately, the pulse rate for detecting individual photons is extremely high and highly variable due to photon shot noise, so QIS systems generally output their captured data as a series of images, although the integration may be somewhat more sophisticated than just a simple counter per pixel[12].

The other obvious set of alternatives is Address-Event Representation (AER) sensors, of which there are several different types[13]. Event sensors differ from spiking sensors in that most incorporate the concept of only emitting a record of data when a pixel's expected value has changed. Generally, this change is detected in the analog domain, which commonly implies that the sensels are conventional analog charge collectors (operating in the photovoltaic mode). The main problem with such sensors is that there is no obvious way to recover the absolute values of pixels from a sequence of deltas without a known reference. Additionally, the charge storage basis imposes dynamic range limits similar to those experienced using conventional sensels.

The point of the current work is that there always was an easy way to build a true IMEV TDCI sensor with the originally desired properties by using the fact that a Light Emitting Diode (LED) is actually a PN junction that can be used precisely the same way we were wanting to use solar cells. The following sections discuss the theory behind use of LEDs as IMEV TDCI sensels, the construction of prototype cameras using low-resolution LED arrays, and summarize the findings.

LED arrays as image sensors

It has long been known that all diodes respond to light to some degree, but the fact that LEDs in particular can be used as light detectors continues to surprise people. Forrest Mims III made this fact common knowledge in the 1970s[15], but perhaps the best known discussion of this use of LEDs now is from 2003[14]. That work explained how a single LED could serve multiple purposes, ranging from a LED backlight that automatically adjusts brightness by detecting the ambient light level to serving as a bidirectional data port that cheaply solves the "last centimeter" problem for low-bandwidth communication with consumer devices. In 2019, that work was significantly expanded upon to include use of LEDs as temperature sensors and a GitHub archive of Arduino code for each application[16].

There are essentially four different procedures by which a LED can be used for light emission or detection:

Light: A LED is normally used as a light source, and this can be done by holding the anode high and the cathode low as shown in Figure 2. A minor detail is that it is usually necessary to limit current by including a resistor in the circuit, most often on the anode. Although the current article is primarily about sensing rather than emitting light, it is possible to freely intermix lighting with any of the sensing procedures. This means that a camera with a LED image sensor also could be used as an image projector, and can even rapidly switch between these functionalities.

Photovoltaic: A LED exposed to light of an appropriate wavelength will generate a flow of electrons from the cathode to the anode. Typical LEDs generate less than 2V with currents in the μA range. Thus, reading the sensel's value requires an analog to digital converter (ADC), as is done in conventional image sensors. A true IMEV sensor would thus require an ADC per sensel, implying a large amount of circuitry. The dynamic range and resolution are limited by how long one waits between readings and by the ADC range and accuracy.

Photoconductive, Wait High: When a LED is held in reverse bias, as shown in Figure 2, the junction capacitance is charged to the high voltage. This charging time is relatively short if resistance is kept low. If the anode is then switched from a low voltage to a high-impedance input, the wait high configuration shown in Figure 2, then photons being absorbed by the LED will cause the charge to leak back to the anode. Over time, the voltage of the anode will thus approach the high voltage applied to the cathode. Rather than measuring the voltage, a simple digital input can be used to time how long it takes for the voltage to reach the threshold level detected as logic 1.

Photoconductive, Wait Low: This process is very similar to that of Photoconductive, Wait High, and also begins with reverse-bias

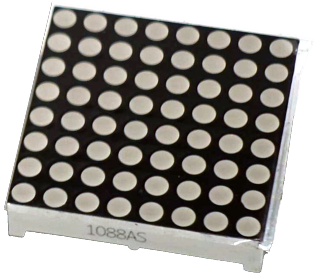

Figure 3. *A 32mm square 8x8 LED array*

charging of the junction capacitance. However, as shown in Figure 2, wait low places the high-impedance input on the cathode rather than the anode. The result is timing loss of the charge on the capacitor rather than the transfer of charge to the anode. Again, a digital input can be used, in this case detecting when the cathode has crossed the threshold level to be read as logic 0.
The ability to do all signal handling in the digital domain is huge advantage for the photoconductive measurement procedures, but it is perhaps more significant that dynamic range and resolution are not constrained by ADC accuracy and a dark sensel can be given arbitrarily long to transfer charge from the junction capacitance. It is also particularly convenient that most microcontrollers allow I/O pins to be switched between a logic output and a suitably high-impedance digital input. However, there is a potential issue in that the operating voltage and voltage thresholds for a given pairing of LED and microcontroller may be less than ideal. LED operating voltages are generally between 1.5V and 4V, whereas microcontrollers most often operate using either 3.3V or 5V, and the precise voltage logic thresholds used may also vary. There is also the potential for I/O pin and stray capacitances to significantly alter photoconductive measurements.

LED packaging

While it is certainly possible to mount an array of individual LEDs for use as an IMEV TDCI sensor, there are also LEDs packaged together. Perhaps the most common single-package-multiple-LED configuration is the tri-color LED. Within a typical four-wire tri-color LED package, there are separate red, green, and blue LEDs with either all anodes or all cathodes tied together on a single pin. This is not particularly problematic. A common cathode is trivially held high throughout the Photoconductive, Wait High procedure. Similarly, a common anode is trivially held low throughout the Photoconductive, Wait Low procedure. Thus, the choice of which is common determines which procedure can be used, but the correct choice gives full functionality.

Perhaps more interesting are the pre-packaged LED arrays. These come in various sizes, but 8x8 arrays of single-color LEDs, like that shown in Figure 3, are very common. The wiring pattern within such a module typically is a grid in which both the anodes and cathodes are shared, each in one dimension. This style of wiring for a 4x4 LED array is shown in Figure 4. This sharing pattern is problematic for direct IMEV TDCI capture.

The connections to the 4x4 matrix are `ABCD1234`, and each connection can be `H` (high), `L` (low), `Z` (disconnected), or `?` (an input). For example, setting all LEDs to reverse bias would be

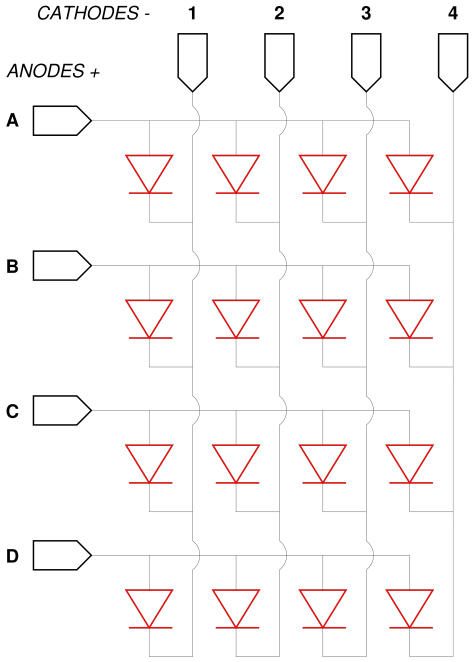

Figure 4. *Inside a 4x4 LED array with common anode/cathode*

`LLLLHHHH`. The problem is that using either wait high or wait low, it is not possible to have all LEDs actively sensing and still perform individual LED reads or resets (to reverse bias). For example, `LLLLHHHH` followed by `????HHHH` could implement wait high, but the LEDs within each row would be sampled as a group. It is possible to individually read the LEDs in one column (by wait high) or one row (by wait low) by only reverse biasing the selected set, but that would essentially produce a rolling shutter effect and would still require 4 LEDs to be reset together. It also is possible to reverse-bias and reset or read a single LED at a time, but this magnifies the rolling shutter artifacts. Using the IMEV TDCI analysis, it would be possible to dynamically schedule the LED samplings based on expected values (i.e., making frequency of sampling each pixel proportional to the expected time between changes in that LED's reading), but the result would still be that only a fraction of the sensels would be getting sampled at any point in time. The wiring pattern and decoding logic design are free choices for the person designing a LED array, so better interconnection patterns could be used.

It is also worth noting that extremely-fine-pitch LED arrays are now being produced, primarily for small display applications. Various types of reasonably high density OLED displays are ubiquitous. More conventional LED panels for larger video-wall displays tend to be around 1mm pixel pitch, but MicroLED displays are commercially produced with pixel pitches as small as 0.2mm – roughly 20K pixels in the space of a 135-film full frame format sensor. Various companies have been experimenting with LEDs smaller than 1μm[17], primarily for use either in micro displays or optical interconnection networks. In sum, it is possible to build arrays of LEDs with densities comparable to pixels in conventional imaging sensors although the detection efficiency of optimized photodiodes is higher.

LED sensing performance

How good are LEDs as sensels? The quick answer is that it varies wildly depending on the properties of the LED design. There are several issues that can dramatically alter the usability of LEDs as sensels:

Insufficient electrical access: It is particularly common that high-density LED arrays are configured with inappropriate-for-sensing control circuitry. For example, this is the case for most commodity OLED displays.

Color sensitivity: LEDs are nearly single-wavelength emitters, producing light within a band that is typically less than 50nm wide. Unfortunately, the same device properties that determine the emitted wavelength also effectively block detection of significantly lower-energy (longer) wavelengths. Thus, most blue LEDs are completely blind to red or even green light. This can be used to advantage in building a color sensor without filters, but it also means panchromatic sensors are best built from LEDs that emit in the red to yellow range.

Use of phosphors: Although a "white LED" can be made by combining light from red, green, and blue LEDs, phosphors are often used to produce a smoother spectrum. Typically, such a white LED is actually emitting blue light, but a fraction of that undergoes a Stokes Shift lowering photon energy and increasing the wavelength as it is absorbed and re-emitted by the phosphor. These phosphors generally do not "work in reverse," so phosphor-based white LEDs make very poor sensels.

QLEDs (LEDs using quantum dots): Similar to Stokes phosphors, quantum dots can be stimulated by blue LEDs to produce a range of colors with lower wavelengths. Unlike phosphors, quantum dots produce very pure colors dependent on the material used and the crystal size, which is in the nanometer range. However, they also do not "work in reverse," so they kill sensitivity.

Colored lenses: The color emitted by a LED comes primarily from properties of the LED itself, but it is common that LEDs are packaged with lenses colored similar to their emission wavelength. Although the lens color has little effect on sensing light near the emission wavelength, it can significantly reduce the fraction of other wavelength photons reaching the LED as a detector. For example, a red LED with a red lens is likely to be less sensitive to green or blue light than if it was packaged with a clear lens.

Lens shape: Common LEDs usually are packaged inside plastic that forms a convex lens over the junction. However, it is also possible to buy LEDs with a flat top or with a concave top. In measuring the responses of dozens of LEDs, it was immediately clear that the concave top LEDs are the least sensitive. The difference between convex and flat tops was less significant, but flat tops do appear to be slightly better. Note that use as an IMEV TDCI sensor essentially places the LEDs in the focus plane of a lens, so ray entry angles are not as steep as might be encountered when using a LED as an unhoused light sensor.

Lens diffusion: The light-emitting junction in a typical LED is quite small. In fact, most LEDs surround the junction with a reflector to broaden the spot. However, the bright central spot is apparent when looking directly at the LED. Many LEDs thus manipulate the lens to provide a more diffuse spot. A diffused LED lens may look cloudy or have a frosted surface rather than being fully transparent. This diffusion generally seemed to reduce sensitivity by a modest amount.

Figure 5. *Ciro-flex CILKY captures IMEV TDCI data using a LED array*

The bottom line is that, aside from the above systemic issues, we found that even apparently very similar LEDs behaved very differently. This is not really surprising because various different materials are used to make LEDs that emit different colors, and those material choices strongly affect the operating voltage and junction behavior. Overall, the most sensitive LEDs of the several dozen types we tested were yellow emitters with a clear flat-top lens. Not far behind was a red version, which we believe also gives a more panchromatic sensitivity, although our test rig did not directly measure spectral properties.

Some prototype cameras

In order to directly test the functionality of LEDs as IMEV TDCI camera sensors, several LED-sensel cameras were built. It is worth noting that these were not the first cameras our lab built using LED sensels, but that previous attempts were done as not-entirely-successful undergraduate student senior projects. Further, those projects did not evaluate many LEDs nor did they implement IMEV TDCI capture. The two prototypes discussed here both functioned as IMEV TDCI cameras, although not without issues. Both are named CILKY, standing for Continuous Imaging LEDs from KentuckY.

Ciro-flex CILKY

The first prototype, Ciro-flex CILKY, is shown in Figure 5. The idea behind this prototype was to make it feel as much like a "real" camera as possible, despite having insanely low pixel count. With that goal, it seemed most reasonable to think of placing the LED array in the film plane of a conventional camera. To make it easy to fit a reasonable number of LEDs, a medium-format body was the obvious choice. Although the controller could generate a live display from the IMEV TDCI stream, aim-

ing and focusing using such low resolution seemed a bad idea, and an SLR would have its optical finder blacked out for the duration of long IMEV captures. Thus, a 6x6cm twin-lens reflex (TLR) body seemed the ideal choice.

The body modified to create Ciro-flex CILKY is a Ciro-flex model B that was manufactured in Delaware, Ohio around 1947. Although this camera originally sold for around $100 (equivalent to nearly $1,500 in 2026), this particular unit was purchased for the disturbingly low price of $3.99 plus $13.50 shipping. As it arrived, it did require some servicing of the shutter and cleaning of the reflex mirror, but this was easily done. It is a typical TLR for the period, using a pair of Wollensak 85mm $f/3.5$ lenses. The Alphax leaf shutter of the taking lens offers timed speeds from 1/200s to 1/10s plus B and T; the T setting is particularly useful in that we want the shutter locked continuously open for IMEV TDCI capture.

As is common for roll film cameras, this camera has a hinged door to the film compartment. It was a very simple matter to remove the hinge pin and thus remove the back. To replace it, first a simple LED mounting plate was 3D printed to clip onto the film rollers above and below the film exposure mask. After experimenting with that, a few iterations of complete replacements for the original back were 3D printed. The goal of the full back was to make the LED sensor, microcontroller, and an OLED display as unobtrusively integrated as possible.

A trick that we have employed in building many custom cameras is that customized circuit-board-like structures for mounting electrical components can be directly 3D printed as parts of complex structures, but without electrical traces. After parts have been inserted, we use a standard wire-wrap tool to make the necessary electrical connections. Because we are directly wrapping standard pins, rather than oversize wire-wrap socket pins, we ensure a durable connection by soldering over the wire-wrapped mechanical connections. Thus, the 3D-printed back has sockets for an ESP32 WROOM microcontroller boardlet and an I2C-interfaced SSD1306 128x64 pixel 0.96" monochrome OLED display to provide the live view. A second 3D-printed component, which clips into the printed back, has holes for mounting a 4x4 array of 5mm LEDs. The total cost of all electrical components is well under $10. The inside of the camera back is shown in Figure 6.

Most of our testing was done with yellow, clear flat-top, 5mm LEDs in the 3D-printed mounting plate, and the system worked very well using that – until the plate was mounted in the camera. The inconsistent results produced with the LED plate in the camera were difficult to diagnose, but the cause was simple: the all-metal camera body and slightly sloppy wire-wrap cabling was causing stray capacitances that, when a human was handling the camera, often dwarfed the junction capacitances we were essentially trying to measure. The prototype with the full back is not only much more neatly wired with a common cathode tied high, but also now has a ground wire to the camera chassis.

Using red, clear flat-top, 5mm LEDs, the Ciro-flex CILKY 260221 version still has some sensitivity to stray capacitance. However, wait high times span a dynamic range of over 18 stops, ranging from tens of microseconds to over 3 seconds. In sum, Ciro-flex CILKY is not a useful camera, but it does prove that LED sensel IMEV TDCI capture is feasible using cheap commodity components.

Figure 6. *Inside Ciro-flex CILKY*

M42 CILKY

Given that Ciro-flex CILKY's metal camera body was a source of stray capacitance and that focusing did not matter much at such low resolutions, we constructed a very minimalist CILKY that would allow mounting a standard M42 lens in front of the sensor without using any metal. M42 is the "universal screw-thread mount" with a diameter of 42mm, 1mm pitch, and a rear focus distance of 45.46mm.

M42 CILKY is shown in Figure 7. The entire body is literally a single 3D-printed part with mounting holes for LEDs on the rear plate, an M42 thread on the front, and an Arca-Swiss-compatible tripod mount underneath including three 1/4-20 threaded tripod mount sockets. This camera's sensor is a 4x4 array of red, clear convex-top, 3mm LEDs. M42 lenses are generally designed to cover a 36x24mm film plane, and thus have an image circle of at least 43.26mm fitting a square image of 30.59mm on a side. Thus, M42 CILKY's 16 LEDs are evenly distributed in a grid on 6mm centers.

Wire-wrapping and soldering connections to a densely-packed array of 3mm LEDs proved challenging, and certainly the blob of hot-melt glue securing the pair of ribbon cables at the

Figure 7. *M42 CILKY: left: without lens, middle: with 55mm f/1.4 lens but no controller, right: projecting*

rear is not pretty. However, the two sets of 9 wires each provide breadboard-compatible connections to each of 16 LED anodes and 2 wires connected to the common cathode with relatively minimal stray capacitance. As shown in Figure 7, not only can this configuration be used to capture 16-pixel data through the 55mm f/1.4 Mamiya/Sekor M42 lens attached, but it also can be used to project an image of that resolution.

A minor detail is that projecting an image using LEDs generally requires a current-limiting resistor, which is traditionally placed on the anode. However, a resistor is not strictly needed if the LED is used only for reverse biasing and sensing. Using a resistor during reverse biasing and sensing slightly reduces the efficiency of the sensing process.

CILKY software

The microcontroller used in both versions of CILKY is the same: an ESP32 WROOM microcontroller boardlet and an I2C-interfaced SSD1306 128x64 pixel 0.96" monochrome OLED display. The software is also the same code written using the ESP32 Arduino environment.

The CILKY software continuously polls all 16 LEDs, asynchronously sampling and resetting: the sampling process is continuous and frameless. The ESP32 is a dual-core 240MHz system, so one core can be dedicated to that real-time process and is more than fast enough so that no external FPGA or other circuitry is needed. Each time a LED hits threshold, the time in microseconds since that LED last hit threshold is computed and recorded as the new current value.

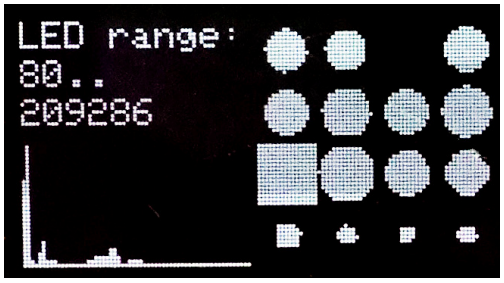

Figure 8. *Ciro-flex CILKY's 128x64 pixel 0.96" OLED display*

The IMEV TDCI data is simply a static initial image followed by a stream of spatio-temporal distances between LED sensel events. The events are filtered by the expected values: an event which produced a sensel value within range of the expected value for that sensel is simply ignored. Although we did not create an empirically-grounded noise model to define the acceptable range around expected values for our LED sensors, the process for computing a high-quality pixel value error PDF (probability density function) is straightforward[18]. For events that are outside the expected range, the distance is computed as:

$$((t - t_{last}) * (X * Y)) + (x - x_{last}) + ((y - y_{last}) * X)$$

Here, X and Y, the dimensions of the LED array, are each 4. The current time (in microseconds) is t and the time at which any sensel last triggered is t_{last}. Similarly, x,y are the coordinates of the currently triggering sensel and x_{last},y_{last} are the coordinates of the previously triggered sensel. This distance is then followed by the actual value of the sensel.

Very simple compression is applied to both the distance and the sensel value. Each number is encoded as a sequence of one or more 8-bit bytes such that the 7 least-significant bits of the number become the bottom 7 bits of the output byte. The 8^{th} bit is coded as 0 if there are no other 1 bits in the number; otherwise, the 8^{th} bit is coded as 1 and the process is repeated with the number shifted right by 7 bit positions. Obviously, a more sophisticated compression scheme could be used, but this is computationally cheap for a microcontroller to do and is one of the standard IMEV stream formats understood by the `tik`[1] and `nutik`[8] TDCI processing software.

As shown in Figure 8, CILKY's OLED display gives the real-time status of the LED sensor. In the upper left, the current range of sensel wait high times is displayed. Below that is a live, autoscaled, incrementally updated histogram of the times. To the right is the live view from the sensor. Because the OLED display offers only two brightness levels, the live view uses the current expected sensel values to select a center-weighted pattern approximating the gray level. Updates happen quickly enough so that human vision averages these patterns somewhat.

Conclusion

The main contribution of this paper is that, using commodity LEDs as sensels, we have empirically proven the feasibility of the type of IMEV TDCI sensor originally proposed in 2014[2]. Timing threshold events to generate an IMEV TDCI data stream is a viable alternative to using ADCs to capture images.

As sensels, LEDs are certainly not ideal, but red or yellow LEDs with clear, flat, tops typically perform better than other options. The reverse bias, wait high sampling procedure was most effective. This approach allows very flexible sampling because each sensel can be independently reset. It also greatly simplifies processing, multiplexing, and compression because all values are inherently digital. The sensels also consume very little power. Despite all the imperfections of our prototype LED cameras, we were able to achieve up to 18 stop dynamic range, which is well beyond what is delivered using conventional ADC-based sensels with a fixed integration (exposure) period.

The most serious impediment to using LED sensels is the extreme sensitivity to stray capacitance. This issue severely penalizes construction methods like the wire-wrap connections used in the CILKY cameras. We plan to construct cameras using larger arrays of LEDs physically configured in ways that should improve immunity to stray capacitance.

The other clear disadvantage is the long time-to-threshold for LED sensels in poor lighting. One can imagine an IMEV quality metric measuring scene information acquired per photon (akin to quantum efficiency for a conventional sensel), and commodity LEDs would not be competitive with sensels designed for sensing. It is not clear how much more competitive a PN junction optimized for *both* emission and sensing could be.

It is worth noting that using LEDs as sensels does not prevent them from being used as light emitters. The two modes cannot be engaged simultaneously, but the devices, wiring, and control logic are easily compatible with rapidly switching between sensel and emitter modes. This was demonstrated with one of the prototype cameras, M42 CILKY.

Various materials about this work and the Ciro-flex CILKY and M42 CILKY prototype cameras are available from:

`https://aggregate.org/DIT/CILKY`

References

[1] Henry Dietz, Paul Eberhart, John Fike, Katie Long, Clark Demaree, Jong Wu, "TIK: a time domain continuous imaging testbed using conventional still images and video" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Digital Photography and Mobile Imaging XIII, 2017, pp 58 - 65, doi: 10.2352/ISSN.2470-1173.2017.15.DPMI-081

[2] Henry Gordon Dietz, "Frameless, time domain continuous image capture," Proc. SPIE 9022, Image Sensors and Imaging Systems 2014, 902207 (4 March 2014), doi: 10.1117/12.2040016

[3] Henry Dietz, Zachary Snyder, John Fike, Pablo Quevedo;, "Scene Appearance Change As Framerate Approaches Infinity" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Digital Photography and Mobile Imaging XII, 2016, doi: 10.2352/ISSN.2470-1173.2016.18.DPMI-259

[4] Henry Dietz, Paul Eberhart, John Fike, Katie Long, Clark Demaree, "Temporal super-resolution for time domain continuous imaging" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Computational Imaging XV, 2017, pp 87 - 93, doi: 10.2352/ISSN.2470-1173.2017.17.COIMG-430

[5] Henry Dietz, Paul Eberhart, and Clark Demaree, "Multispectral, high dynamic range, time domain continuous imaging" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Photography, Mobile, and Immersive Imaging, 2018, pp 409-1 - 409-9, doi: 10.2352/ISSN.2470-1173.2018.05.PMII-409

[6] Henry Dietz, Clark Demaree, Paul Eberhart, Chelsea Kuball, and Jong Yeu Wu, "Lessons from design, construction, and use of various multicameras" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Photography, Mobile, and Immersive Imaging, 2018, pp 182-1 - 182-10, doi: 10.2352/ISSN.2470-1173.2018.05.PMII-182

[7] Paul Eberhart and Henry G. Dietz, "Non-Uniform Integration of TDCI Captures" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Imaging Sensors and Systems, 2020, pp 330-1 - 330-7, doi: 10.2352/ISSN.2470-1173.2020.7.ISS-330

[8] Paul Eberhart and Henry G. Dietz, "NUTIK: A testbed for functional post-capture manipulation of time and gain" in Proc. IS&T Int'l. Symp. on Electronic Imaging: Computational Imaging, 2026

[9] W. Yang, "A wide-dynamic-range, low-power photosensor array," in Proceedings of IEEE International Solid-State Circuits Conference - ISSCC, pp. 230-231, 1994, doi: 10.1109/ISSCC.1994.344657

[10] Xin Qi, Xiaochuan Guo, and John G. Harris, "A time-to-first spike CMOS imager" in 2004 IEEE International Symposium on Circuits and Systems (ISCAS), vol. 4, pp. IV-824, 2004

[11] Eric R.Fossum, Jiaju Ma, Saleh Masoodian, Leo Anzagira, and Rachel Zizza "The Quanta Image Sensor: Every Photon Counts" in Sensors 16, no. 8: 1260, 2016, doi: 10.3390/s16081260

[12] Chan, Stanley H., Omar A. Elgendy, and Xiran Wang, "Images from Bits: Non-Iterative Image Reconstruction for Quanta Image Sensors" in Sensors 16, no. 11: 1961, 2016, doi: 10.3390/s16111961

[13] T. Delbruck, B. Linares-Barranco, E. Culurciello and C. Posch, "Activity-driven, event-based vision sensors" in Proceedings of 2010 IEEE International Symposium on Circuits and Systems, Paris, France, 2010, pp. 2426-2429, doi: 10.1109/ISCAS.2010.5537149

[14] Paul Dietz, William Yerazunis, and Darren Leigh, "Very Low-Cost Sensing and Communication Using Bidirectional LEDs" in Dey, Schmidt, and McCarthy (eds) UbiComp 2003: Ubiquitous Computing, Lecture Notes in Computer Science, vol 2864, Springer, Berlin, Heidelberg, 2003, doi: 10.1007/978-3-540-39653-6_14

[15] Forest M. Mims III, *LED Circuits & Projects*, Howard W. Sams and Co. Inc., New York, pp. 60-61, 76-77, and 122-123, 1973

[16] Paul H. Dietz and Jennifer (Ginger) Alford, "LEDs as sensors" isbn 9781450363167, Association for Computing Machinery, New York, NY, USA, 2019, doi: 10.1145/3306306.3328752, github url: `https://github.com/paulhdietz/LEDSensors`

[17] Rahul Rao, "LEDs Enter the Nanoscale" IEEE Spectrum News, February 12, 2026, url: `https://spectrum.ieee.org/nanoled-research-approaches`

[18] Henry Gordon Dietz, "Construction, Quality Assessment, and Applications of Pixel Value Error PDF Models" in Electronic Imaging, 2025, pp 225-1 - 225-11, doi: 10.2352/EI.2025.37.10.IPAS-225