ESIM Simulator and Datasets

Research on event cameras and event-based computer vision is still in its early stages. One of these reasons is tied to the hardware itself, since event cameras are expensive, not widely available, and are mostly prototypes, with low resolution (no more than 346x260 in the case of a DAVIS346) and therefore not ready for commercial applications. Another such reason is the need for data and datasets, which are also scarce for the time being.

To tackle these problems, a number of simulators have been developed, of which ESIM [Rebecq18] is an example, developed by Depts. Informatics and Neuroinformatics at ETH Zurich, in a similar spirit as simulator for conventional cameras. ESIM is an open-source ROS package.

Simulators and ESIM

These simulators continuously render images, which are used to generate the events, based on differences between frames.

A common problem with these simulators is the choice of framerate, as a naïve approach would have the engine render at 1000 frames per second, in order to match the microsecond temporal resolution of event cameras. ESIM uses a tightly coupled system between the simulator and the rendering engine, allowing for a better simulation of event cameras, by using an adaptive sampling based on the dynamics of the scene.

A picture containing different, sitting, various, water

Description automatically generated

Figure 1 Reference image [Rebecq18]

A close up of a map

Description automatically generated

Figure 2 Comaprison of uniform vs adpative sampling [Rebecq18]

This adaptive sampling is possible due to a communication between the simulator itself, and the rendering engine being used. It relies on the trajectory of the virtual camera, as well as changed due to brightness, pixel displacement (and interpolation between images for motion estimation), and noise and non-idealities.

A screenshot of a cell phone

Description automatically generated

Figure 3 Simulator architecture [Rebecq18]

ESIM allows the use of multiple rendering engines and modes, such as as an OpenGL integration, as well as a photorealistic rendering using Unreal Engine. The simulation of stereo cameras, and panoramic cameras is also possible. Furthermore, it is possible to “convert” videos to events, though the results are not as good, due to the fixed sampling rate.

Datasets (http://rpg.ifi.uzh.ch/davis\_data.html)

Datasets provide an interesting way to test algorithms without the need for a camera, compare results with other algorithms, and have a ground truth with which to benchmark.

The datasets I have used are shapes\_6dof.bag, which contains footage from a DAVIS240 of a collection of geometric shapes on a wall, and boxes\_6dof.bag, which contains footage from a DAVIS240 of multiple boxes, with a complex texture.

Simulator usage

Integrar esta parte com o que esta escrito no tst\_sw, e por imagens das diretorias

References

[Rebecq18] : ESIM: an Open Event Camera Simulator