LEARNING TO RECONSTRUCT INTENSITY IMAGES FROM EVENTS

OLAYLARDAN YEĞİNLİK GÖRÜNTÜLERİ GERİÇATMAYI ÖĞRENMEK

BURAK ERCAN

PROF. DR. MEHMET ERKUT ERDEM

Supervisor

ASSOC. PROF. DR. İBRAHİM AYKUT ERDEM

2nd Supervisor

Submitted to

Graduate School of Science and Engineering of Hacettepe University

as a Partial Fulfillment to the Requirements

for the Award of the Degree of Doctor of Philosophy

in Computer Engineering

May 2024

This work titled **"LEARNING TO RECONSTRUCT INTENSITY IMAGES FROM EVENTS"** by **Burak Ercan** has been approved as a thesis for the Degree of **DOCTOR OF PHILOSOPHY IN COMPUTER ENGINEERING** by the below mentioned Examining Committee Members.

Prof. Dr. Nazlı İKİZLER CİNBİŞ
Head . . . . . .

Prof. Dr. Mehmet Erkut ERDEM
Supervisor . . . . . .

Prof. Dr. Aydın ALATAN
Member . . . . . .

Assoc. Prof. Dr. Emre AKBAŞ
Member . . . . . .

Assoc. Prof. Dr. Hacer YALIM KELEŞ
Member . . . . . .

This thesis has been approved as a thesis for the Degree of **DOCTOR OF PHILOSOPHY IN COMPUTER ENGINEERING** by Board of Directors of the Institute for Graduate Studies in Science and Engineering on ... / ... / .....

Prof. Dr. Benat KOÇKAR
Director of the Institute of
Graduate School of Science and Engineering

To my lovely wife and daughter.

ETHICS

In this thesis study, prepared in accordance with the spelling rules of Institute of Graduate Studies in Science of Hacettepe University,

I declare that

- all the information and documents have been obtained in the base of the academic rules.
- all audio-visual and written information and results have been presented according to the rules of scientific ethics
- in case of using others works, related studies have been cited in accordance with the scientific standards
- all cited studies have been fully referenced
- I did not do any distortion in the data set
- and any part of this thesis has not been presented as another thesis study at this or any other university.

... / ... / .....

Burak Ercan

YAYINLAMA FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan **"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"** kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- ☐ Enstitü yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir.
- ☐ Enstitü yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren .... ay ertelenmiştir.
- ☐ Tezim ile ilgili gizlilik kararı verilmiştir.

... / ... / .....

Burak Ercan

ABSTRACT

LEARNING TO RECONSTRUCT INTENSITY IMAGES FROM EVENTS

Burak ERCAN

Doctor of Philosophy, Computer Engineering
Supervisor: Prof. Dr. Mehmet Erkut ERDEM
2nd Supervisor: Assoc. Prof. Dr. İbrahim Aykut ERDEM
May 2024, 229 pages

The past decade has seen significant progress in computer vision, leading to diverse applications across various domains. However, today's artificial vision systems remain in their infancy compared to their biological counterparts in robustness to challenging real-world scenarios, real-time processing capabilities, and computational efficiency. These shortcomings can be attributed to the classical frame-based acquisition and processing pipelines, which suffer from low temporal resolution, low dynamic range, motion blur, and redundant information flow.

A new class of visual sensory devices called event cameras offers promising solutions to these challenges. Instead of capturing frames collectively, the pixels of an event camera work independently and respond to local brightness variations by generating asynchronous signals called events. As a result, event cameras have many advantages over traditional frame-based

sensors, such as high dynamic range, high temporal resolution, low latency, and minimal motion blur.

This thesis focuses on reconstructing intensity images from events. Reconstruction of intensity information leverages the advantages of events for high-quality imaging in challenging scenarios. This enables the application of established methods developed for frame-based images and facilitates human-centered applications involving event data. We present three main contributions on this task: a novel method surpassing existing ones in terms of image quality and efficiency, a comprehensive evaluation framework, and a large and diverse benchmark dataset.

First, we develop a novel dynamic neural network architecture based on hypernetworks, named HyperE2VID. HyperE2VID dynamically adapts to event data, unlike existing works that process events with static networks. Its context fusion module leverages complementary elements of event and frame domains, while its filter decomposition steps reduce computational cost. Thanks to this design, it surpasses existing methods in both image quality and computational efficiency.

Our second contribution is an open-source library for evaluating and analyzing event-based video reconstruction methods, called EVREAL. EVREAL allows us to evaluate different methods comprehensively, considering diverse and challenging scenarios, employing extensive real-world datasets, measuring robustness to several key variables, and assessing performance through multiple metrics and tasks. This evaluation ensures generalizability to real-world scenarios, fair comparison, and reproducibility.

Our third contribution is a new benchmark dataset, HUE. HUE has high resolution, contains numerous sequences taken in diverse scenarios, and focuses on low-light scenarios, a challenging but rewarding domain for event-based video reconstruction.

Using EVREAL, we evaluate HyperE2VID via extensive experiments on several datasets, including our proposed dataset HUE. We use various metrics to assess the image quality under different conditions. We also analyze computational complexity and present a detailed

ablation study to validate the design choices of HyperE2VID. Our experimental results demonstrate the success of the proposed dynamic architecture, generating higher-quality videos than previous state-of-the-art methods across a wide range of settings, while also reducing memory consumption and inference time.

We expect the event-based vision literature to keep growing and event cameras to become more prominent in the coming years. We believe our method HyperE2VID, together with our evaluation framework EVREAL and benchmark dataset HUE, marks an important step towards enabling high-quality and robust imaging in a computationally efficient way.

Keywords: Event Cameras, Dynamic Vision Sensor, Event-based Vision, Video Reconstruction

ÖZET

OLAYLARDAN YEĞİNLİK GÖRÜNTÜLERİ GERİÇATMAYI ÖĞRENMEK

Burak ERCAN

Doktora, **Bilgisayar Mühendisliği**
Tez Danışmanı: Prof. Dr. Mehmet Erkut ERDEM
Eş Danışman: Doç. Dr. İbrahim Aykut ERDEM
Mayıs 2024, 229 sayfa

Son on yılda bilgisayarlı görme alanında önemli ilerlemeler kaydedilmiş ve pek çok alanda çeşitli uygulamaların yolu açılmıştır. Yine de, günümüzün yapay görme sistemleri, zorlu gerçek dünya senaryolarına olan dayanıklılıkları, gerçek zamanlı işleme kapasiteleri ve hesaplama verimliliği açısından biyolojik benzerlerine kıyasla hala emekleme aşamasındadır. Bu durumun önemli sebeplerinden biri; düşük zamansal çözünürlüğe, düşük dinamik aralığa, hareket bulanıklığına ve gereksiz bilgi akışına neden olan klasik çerçeve tabanlı görüntü alma ve işleme yöntemlerini izlemeleridir.

Olay kameraları adı verilen yeni bir görsel algılayıcı sınıfı, bu sorunlara umut verici çözümler sunmaktadır. Bir olay kamerasının pikselleri, beraberce bir çerçeve yakalamak yerine bağımsız olarak çalışır ve yerel parlaklık değişikliklerine yanıt olarak olay adı verilen asenkron sinyaller üretirler. Bunun sonucunda olay kameraları, geleneksel çerçeve tabanlı

sensörlere kıyasla yüksek dinamik aralık, yüksek zamansal çözünürlük, düşük gecikme süresi ve minimal hareket bulanıklığı gibi birçok avantaja sahiptir.

Bu tez, olaylardan yeğinlik görüntüleri geriçatmaya odaklanmaktadır. Yeğinlik bilgisinin yeniden oluşturulması, zorlu senaryolarda olayların avantajlarından yararlanarak yüksek kaliteli görüntüleme sağlamaktadır. Bu da çerçeve tabanlı görüntüler için geliştirilen mevcut yöntemlerin uygulanabilmesini sağlar ve olay verilerini içeren insan merkezli uygulamaları kolaylaştırır. Bu göreve ilişkin üç ana katkı sunmaktayız: görüntü kalitesi ve verimlilik açısından mevcut yöntemleri aşan yenilikçi bir yöntem, kapsamlı bir değerlendirme kütüphanesi, ve geniş bir denektaşı veri kümesi.

Öncelikle, HyperE2VID adında, hiperağlara dayanan, yenilikçi ve dinamik bir sinir ağı mimarisi geliştirilmiştir. HyperE2VID, olay verilerini statik ağlarla işleyen mevcut çalışmaların aksine, olay verilerine dinamik olarak uyum sağlamak üzere tasarlanmıştır. Kullanılan bağlam birleştirme modülü, olay ve çerçevelerin birbirini tamamlayıcı unsurlarından yararlanmakta, filtre ayrıştırma adımları ise hesaplama maliyetini azaltmaya yardımcı olmaktadır. Bu tasarım sayesinde, HyperE2VID hem görüntü kalitesi hem de hesaplama verimliliği açısından mevcut yöntemleri aşmaktadır.

İkinci katkımız, olaylardan yeğinlik geriçatma yöntemlerini değerlendirmek ve analiz etmek için açık kaynaklı bir kütüphane olan EVREAL'dır. EVREAL, çeşitli zorlu senaryoları dikkate alarak, kapsamlı gerçek dünya veri kümelerini kullanarak, birkaç anahtar değişkene karşı sağlamlığı ölçerek ve birden çok metrik ve görev aracılığıyla performansı değerlendirerek, farklı geriçatma yöntemlerini bütünleşik ve kapsamlı bir şekilde değerlendirebilmemizi sağlamaktadır. Bu değerlendirme gerçek dünya senaryolarına genelleştirilebilirliği, adil karşılaştırmayı ve yeniden üretilebilirliği garanti etmektedir.

Üçüncü katkımız, HUE adında yeni bir denektaşı veri kümesidir. HUE veri kümesi yüksek çözünürlüğe ve çeşitli senaryolarda çekilmiş çok sayıda sekansa sahiptir. Önemli bir kısmı olaylardan yeğinlik görüntüleri geriçatma açısından zorlayıcı ancak ödüllendirici olan düşük ışıklı sahneleri içerecek şekilde çekilmiştir.

HyperE2VID, önerdiğimiz HUE veri seti de dahil olmak üzere çeşitli veri setlerinde ve EVREAL kullanılan geniş çaplı deneylerle değerlendirilmiştir. Her yöntemin farklı koşullar altındaki görüntü kalitesini değerlendirmek için çeşitli metrikler kullanılmıştır. Ayrıca hesaplama karmaşıklığı analiz edilmiş ve HyperE2VID'in tasarım seçimlerinin birçoğunu doğrulamak için detaylı bir ablasyon çalışması sunulmuştur. Deneysel sonuçlar, yaklaşımımızın görsel kalite açısından önceki yöntemlere göre daha iyi videolar oluşturabildiği ve aynı zamanda daha düşük bellek tüketimine sahip olduğu ve çıkarım sürelerini azalttığını göstererek önerilen dinamik mimarinin başarısını vurgulamıştır.

Olay tabanlı görme literatürünün büyümeye devam edeceğini ve olay kameralarının önümüzdeki yıllarda daha yaygın hale geleceğini öngörmekteyiz. Yöntemimiz HyperE2VID'in, değerlendirme kütüphanemiz EVREAL ve denektaşı veri kümemiz HUE ile birlikte, hesaplama açısından verimli şekilde yüksek kaliteli ve gürbüz görüntüleme sağlamaya yönelik önemli bir adım olduğuna inanıyoruz.

Anahtar Kelimeler: Olay Kameraları, Dinamik Görme Sensörü, Olay Tabanlı Görme, Görüntü Geriçatma

ACKNOWLEDGEMENTS

First, I would like to thank my supervisors, Prof. Dr. Erkut Erdem and Assoc. Prof. Dr. Aykut Erdem, for their guidance throughout this long journey. Their knowledge, experience, and support were indispensable for this thesis to become a success, and I am grateful to have had them as my advisors.

I thank my thesis advisory committee members, Prof. Dr. Nazlı İkizler Cinbiş and Assoc. Prof. Dr. Emre Akbaş, for their helpful comments, feedback, and guidance. I would also like to thank the jury members, Prof. Dr. Aydın Alatan and Assoc. Prof. Dr. Hacer Yalım Keleş, for reviewing this thesis and providing insightful comments.

I thank my co-authors, Onur and Canberk, for all the hard work they have put in and for bearing with me through all those long discussions. I thank all my professors, colleagues, friends, and anyone who helped me become who I am and get to where I am now.

My biggest thanks go to my devoted and beautiful wife, Gülşah. She believed in me, and supported me through it all, with her never-ending love and patience. This thesis would not have been possible without her. Our daughter Arya brought even more love, joy, and meaning to our lives. I am incredibly grateful for having her and will always love her.

I want to thank my mother, Çiğdem, and father, Hürer, for all their efforts and support in helping me get to where I am today. I sincerely thank my parents-in-law, Aysel and Yusuf Özdemir, for the immense support they provide whenever we need it. I would also like to thank my brother, Buğra, for all his help and support, and for always being there for me.

This thesis is supported in part by the Scientific and Technological Research Council of Turkey (TÜBİTAK) by 1001 Program Award No. 121E454, the Hacettepe University Scientific Research Projects (BAP) Coordination Unit under Grant FHD-2023-20611, the Koç University & İş Bank Artificial Intelligence Center (KUIS AI) Research Award, and the Young Scientist Awards Program (BAGEP) 2021 Award of the Science Academy to Aykut Erdem.

CONTENTS

Page

TABLES

FIGURES

ABBREVIATIONS

ACDA	:	**A**daptive **C**onvolutions with **D**ynmaic **A**toms
AER	:	**A**ddress **E**vent **R**epresentation
AMP	:	**A**daptive **M**embrane **P**otential
ANN	:	**A**rtificial **N**eural **N**etwork
AP	:	**A**verage **P**recision
AR	:	**A**ugmented **R**eality
ATIS	:	**A**synchronous **T**ime-based **I**mage **S**ensor
BRISQUE	:	**B**lind/**R**eferenceless **I**mage **S**patial **Qu**ality **E**valuator
BS-ERGB	:	**B**eam **S**plitter **E**vent **RGB**
CED	:	**C**olor **E**vent **D**ataset
CF	:	**C**ontext **F**usion
CL	:	**C**urriculum **L**earning
CLIP	:	**C**ontrastive **L**anguage-**I**mage-**P**retraining
CMOS	:	**C**omplementary **M**etal-**O**xide-**S**emiconductor
CNN	:	**C**onvolutional **N**eural **N**etwork
CPU	:	**C**entral **P**rocessing **U**nit
DAVIS	:	**D**ynamic and **A**ctive-pixel **VI**sion **S**ensor
DDPM	:	**D**enoising **D**iffusion-**P**robabilistic **M**odels
DFG	:	**D**ynamic **F**ilter **G**eneration
DNN	:	**D**eep **N**eural **N**etwork
DOF	:	**D**egrees **o**f **F**reedom
DVS	:	**D**ynamic **V**ision **S**ensor
E2VID	:	**E**vent to **VID**eo
ECD	:	**E**vent **C**amera **D**ataset
EKF	:	**E**xtended **K**alman **F**ilter
EVG	:	**E**vent **V**oxel **G**rid

FLOP : **F**loating **P**oint **O**perations

FPS : **F**rames **P**er **S**econd

FPV : **F**irst **P**erson **V**iew

FPVDR : **F**irst **P**erson **V**iew **D**rone **R**acing

FSIM : **F**eature **S**imilarity **I**ndex **M**easure

GAN : **G**enerative **A**dversarial **N**etwork

GCN : **G**raph **C**onvolutional **N**etwork

GPIO : **G**eneral **P**urpose **I**nput/**O**utput

GPU : **G**raphics **P**rocessing **U**nit

GRU : **G**ated **R**ecurrent **U**nit

GWD : **G**romov **W**asserstein **D**iscrepancy

HDR : **H**igh **D**ynamic **R**ange

HQF : **H**igh **Q**uality **F**rames

HUE : **H**acettepe **U**niversity **E**vent

ICNR : **I**nitialized to **C**onvolution **N**earest-neighbor **R**esize

IoT : **I**nternet **o**f **T**hings

IQA : **I**mage **Q**uality **A**ssessment

IWE : **I**mage of **W**arped **E**vents

LED : **L**ight-**E**mitting **D**iode

LIF : **L**eaky-**I**ntegrate-and-**F**ire

LPIPS : **L**earned **P**erceptual **I**mage **P**atch **S**imilarity

LSTM : **L**ong **S**hort-**T**erm **M**emory

MANIQA : **M**ulti-dimension **A**ttention **N**etwork for no-reference **I**mage **Q**uality **A**ssessment

MAPE : **M**ean **A**bsolute **P**ercentage **E**rror

MDOE : **M**agnitude and **D**ensity **O**f **E**vents

MLP : **M**ulti **L**ayer **P**erceptron

MP : **M**embrane **P**otential

MSE : **M**ean **S**quared **E**rror

MVSEC : **M**ulti-**V**ehicle **S**tereo **E**vent **C**amera

NIQE : **N**aturalness **I**mage **Q**uality **E**valuator

NODE : **N**eural **O**rdinary **D**ifferential **E**quation

PR : **P**revious **R**econstruction

PSNR : **P**eak **S**ignal-to-**N**oise **R**atio

ReLU : **R**ectified **L**inear **U**nit

RGB : **R**ed, **G**reen, **B**lue

RNN : **R**ecurrent **N**eural **N**etwork

SLAM : **S**imultaneous **L**ocalization **A**nd **M**apping

SNN : **S**piking **N**eural **N**etwork

SPADE : **SP**atially-**A**daptive **DE**-normalization

SPET : **S**ingle **P**ixel **E**vent **T**ensor

SSIM : **S**tructural **S**imilarity **I**ndex **M**easure

SSM : **S**tate **S**pace **M**odels

SVD : **S**ingular **V**alue **D**ecomposition

TORE : **T**ime-**O**rdered **R**ecent **E**vent

UZH : **U**niversity of **Z**urich

ViT : **Vi**sion **T**ransformer

VR : **V**irtual **R**eality

2D : **Two**-**D**imensional

3D : **Three**-**D**imensional

1. INTRODUCTION

Throughout the past decade, the field of computer vision has seen some astonishing achievements in tasks such as image classification [13–15], object detection [16–20], segmentation [21–24], pose estimation [25–29], optical flow estimation [30], visual odometry [31], monocular depth estimation [32–35], object tracking [36–38], video understanding [39, 40] and image synthesis [41–43]; thanks to the recent progress in deep learning methodologies [44] and deep neural network (DNN) architectures such as convolutional neural networks (CNN) [45, 46] and vision transformers (ViT) [47, 48]. This has led to many successful vision applications in different domains, such as medical image analysis [49–53], robotics [54–58], physical reasoning [59, 60], autonomous driving [61–63], remote sensing [64], playing video games [65], and astronomy [66], to name a few. All these advances pave the path for machines that can process visual sensory information to perceive the world as successfully as, if not better than, humans and other biological species with impressive visual systems.

However, despite all these advances, today's artificial vision systems are still falling short for real-world tasks involving high-speed motion, high dynamic range scenes, real-time and low-power processing, etc., compared to their biological counterparts. Even tiny insects perform routine real-world tasks involving real-time perception more successfully than today's computer systems and are much more energy-efficient [67].

Some of the shortcomings of artificial vision systems can be attributed to the classical frame-based acquisition and processing pipelines they follow. This frame-based approach dates back to as early as 1878, when Eadweard Muybridge used a dozen cameras, each triggered sequentially by a set of strings, to capture a series of photographs that depicted the movement of a horse in gallop. His resulting work, *Sallie Gardner at a Gallop*, became the first example of chronophotography and marked an important step in the development of motion pictures (Figure 1.1).

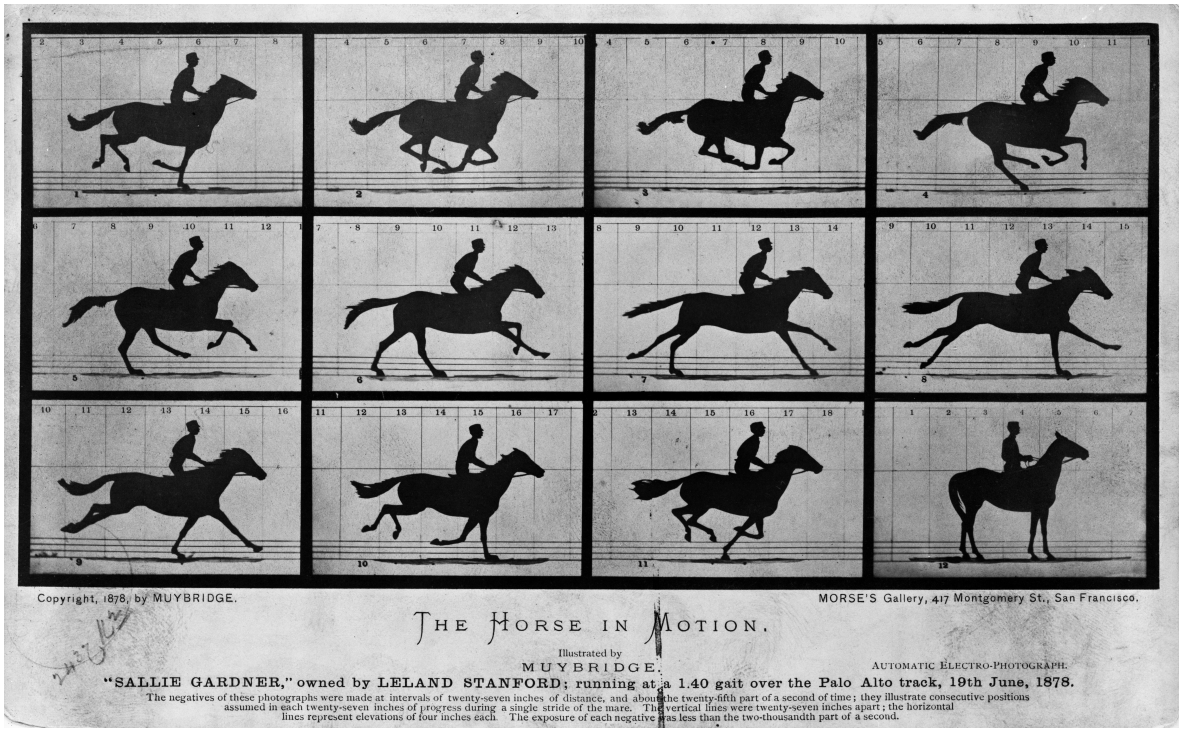


Figure 1.1 On 19th June 1878, the English photographer Eadweard Muybridge managed to capture photographs of a horse named Sallie Gardner, running at full speed. He used a dozen cameras, each triggered sequentially by a set of strings in approximately $40\,\mathrm{ms}$ intervals and using $500\,\mu\mathrm{s}$ exposure times. The resulting work shown in this picture, titled "The Horse in Motion" or "Sallie Gardner at a Gallop", became the first example of chronophotography, and revelaed that the horses had all four feet off the ground at the same time during gallop. Figure from [7].

Today, most artificial vision systems still follow this frame-based approach to capture motion, by relying on visual information acquired by digital frame-based sensors (*i.e.*, rolling or global shutter sensors) in the form of intensity image frames. However, due to their basic principles for collecting visual information, these frame-based sensors have problems, such as motion blur, low temporal resolution, and low dynamic range. Furthermore, sequences of images acquired with these sensors carry too much redundant information since most pixels do not change at every frame. Acquiring, transmitting, and processing this redundant information decreases the efficiency of these systems, and due to this low efficiency, these systems either operate with high latency or require high power consumption [68].

1.1. Event Cameras and Event-based Vision

Another class of visual sensory devices called event cameras, such as DVS [69], ATIS [70], and DAVIS [71], is gaining some popularity recently, and they have the potential to eliminate the problems mentioned above. Event cameras incorporate novel bio-inspired vision sensors, which mimic the information flow in the low-level visual system of mammalians. Like biological retinas, these devices are primarily sensitive to spatio-temporal contrast, coding positive and negative local brightness changes into separate output channels (ON/OFF). Figure 1.2 shows a simplified three-layer model of the human retina and corresponding circuitry in an event camera. Since these sensors are biologically inspired, they are also referred to as retinomorphic sensors, neuromorphic vision sensors, or silicon retinas. [72].

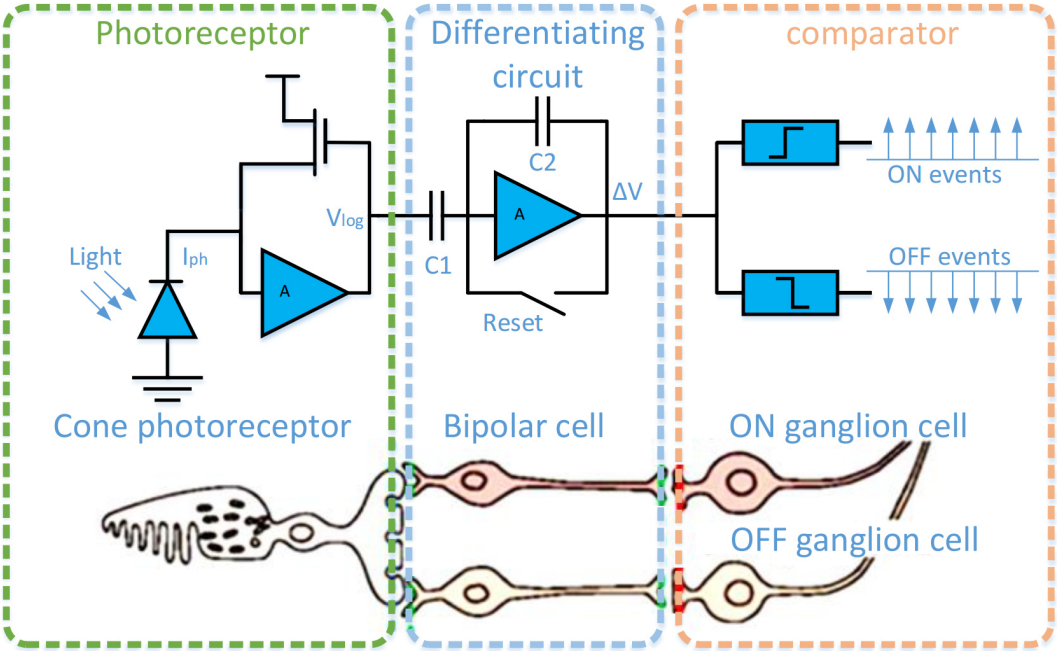


Figure 1.2 A simplified three-layer model of the human retina consisting of photoreceptors, bipolar cells, and ganglion cells (bottom), and the corresponding DVS pixel circuitry implementing a similar flow of visual information (top). Figure taken from [8].

Similar to traditional frame-based imaging sensors, event cameras have two-dimensional pixel arrays with photo-detectors sensitive to electromagnetic radiation in the visible (or infrared) light spectrum. However, in stark contrast to traditional sensors, the pixels of these novel sensors do not integrate light intensity information for a pre-defined amount of time

synchronized to a global clock signal, and their values are not read out collectively as a sequence of two-dimensional frames. Instead, the pixels of event cameras are asynchronous and work independently from each other. Each of the pixels is sensitive to local relative light intensity variations, and when this variation exceeds a threshold, they generate signals called events in continuous time. Therefore, the data output from these cameras is a stream of asynchronous events. Figure 1.3 depicts different outputs between event and standard frame-based cameras.

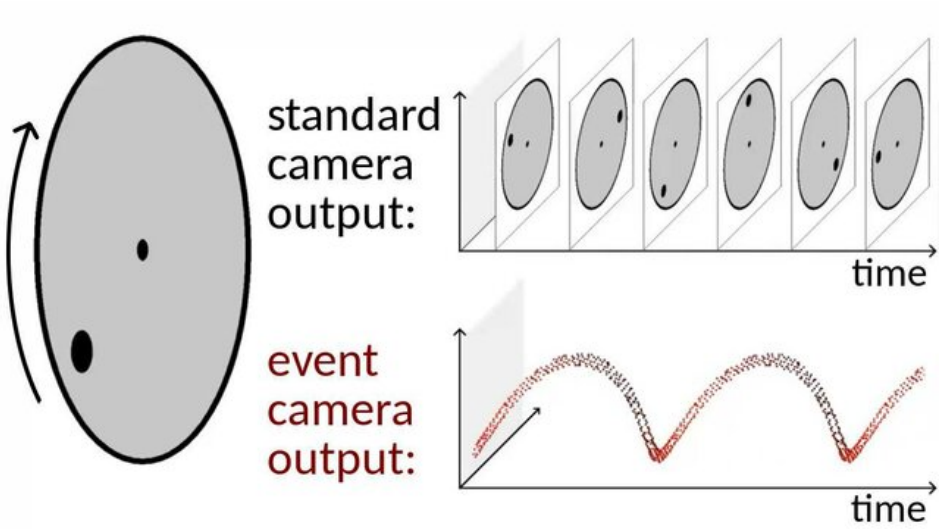


Figure 1.3 Comparison of standard frame-based and event camera outputs. Both cameras observe a scene consisting of a rotating disk with a black dot. Standard camera outputs frames that are equispaced in time, regardless of the scene. They continue to output these frames even when the disk stops. On the other hand, the event camera outputs a continuous stream of events, depicted as points in space-time, triggered by the pixels where intensity levels increase and decrease due to the moving black dot. The output rate depends on the scene; no event (other than noise events) is generated when the disk stops. Figure taken from [9].

Each event $e \doteq (x, y, t, p)$ encodes the pixel location (x, y) and polarity p of the intensity change (*ON* or *OFF*), together with a precise timestamp t. This representation, known as the Address Event Representation (AER), was initially developed to convey the location and timing information of sparse neural events between neuromorphic chips [73] and has become the standard format utilized by event-based sensors since.

These operating principles bring many advantages to event cameras compared to traditional frame-based ones, such as high dynamic range, high temporal resolution, low latency,

and minimal motion blur. We cover these with more detail in Section 2.1.3. Due to these advantages and the increasing availability of prototype or commercial event cameras, there has been a growing research interest from academia and industry in recent years in using event cameras for computer vision tasks that can be difficult with standard frame-based cameras (Figure 1.4). However, since these event-based cameras output spatio-temporal information in a very different form compared to typical frames and have different photometric response and noise characteristics, directly applying methodologies from decades of computer vision research was not possible. This has led to several different event representations, processing methodologies, tasks, and applications, giving birth to a new sub-field called event-based vision. We briefly review the literature from this sub-field in Chapter 2., while more detailed surveys can be found in [74] and [75].

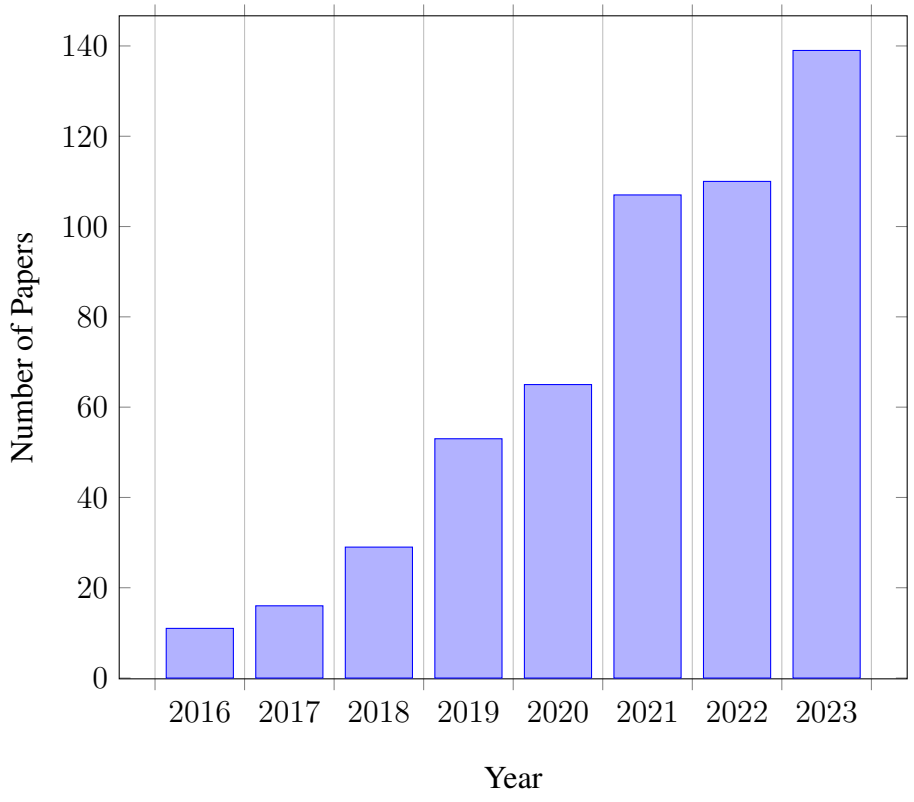


Figure 1.4 A chart showing total number of papers on event-based vision published in prominent computer vision and robotic venues in recent years. Data gathered in [10].

1.2. Task Definition and Motivation

Our primary motivation is to understand the nature of events better and to analyze less explored computational methodologies that can be a better fit to process event data; to obtain novel, efficient, effective, and practical algorithms that can enable the creation of artificial vision sensing and perception that better matches their biological counterparts. Following this primary motivation, we choose intensity image reconstruction as the focus of this study. Thanks to the advantages of event data, it is possible to generate single or sequence of intensity images with desirable properties such as high dynamic range [76], high frame per second [77, 78], minimal motion blur [79–81], and super-resolution [82–84]. These properties enable successful imaging in situations where traditional cameras fail, such as scenes with low light [85, 86] and high-speed motion [87], and enable novel imaging techniques in domains such as microscopy [88] and schlieren imaging [89].

These intensity frames can either be generated using event data only [90, 91] or by fusing event data with complementary lower-quality intensity frames to enhance them [92, 93]. In this thesis, we focus on reconstructing high-quality frames by relying on only events as input, inspired by the efficiency and success of biological vision systems and motivated by the real-world impact of having artificial vision systems with lower cost, size, weight, power, and latency. In the following, we elaborate on the motivations for generating intensity images from events.

First, high-quality intensity images are the most natural way for us humans to visualize and interpret event data. Therefore, intensity image reconstruction from events is indispensable for human-centered applications involving event data.

Second, by reconstructing high-quality intensity images, it becomes possible to directly apply both established and modern successful methods developed for standard frame-based images to downstream tasks. This approach offers numerous benefits and use cases. There are more image datasets than event datasets, and the vast majority of computer vision research is conducted using these abundant frame-based intensity data. By reconstructing

images, successful methods from frame-based vision can be directly applied to intensity reconstructions of events for various tasks, yielding good results with less effort, without the need to develop specialized methods for events. Additionally, employing a modular approach that uses separate modules for intensity reconstruction from events and a frame-based method for a downstream task brings further benefits, such as the ability to replace the second module with improved methods as new research proposes better techniques for frame-based downstream tasks. Furthermore, reconstruction acts as a baseline and shows what performance can be achieved (at the very least) on a specific task with event data [94], motivating the development of more specialized methods with possibly better performance.

Reconstruction also allows one to use established tools and techniques for tasks that are well-studied in the literature. An example of this is the task of geometric camera calibration. In [95], Muglikar *et al*. show that reconstructing intensity images from events is well suited for calibrating event cameras, allowing one to use standard calibration patterns instead of blinking LED patterns or external screens and providing a straightforward method for performing extrinsic calibration between frame-based and event-based sensors.

Intensity images and events are visual modalities that are closely related to each other, and the reconstruction task bridges and brings out the relations between these two modalities. This makes event-based image reconstruction a notable component for research on other event-based vision tasks as well, with use cases ranging from simultaneous estimation with other variables to bringing additional supervision or acting as an intermediate representation.

Starting with the pioneering work of Cook *et al*. [96], simultaneously estimating multiple quantities like intensity images, spatial gradients, and optical flow has proved useful [97–99]. This multi-faceted approach benefits from the dynamic interaction between these elements, as exemplified by the event generation model of Gallego *et al*. [100], which correlates optical flow, scene gradients, and event data. Another benefit of estimating variables together, especially in the deep learning era, is that it allows researchers to make use of these relations to design methods that use less supervision for training, as shown in [101] and [102]. In [103], the authors use an event-based video reconstruction network in their unsupervised

domain adaptation framework, which allows leveraging labeled image datasets to train for unlabeled events in tasks like semantic segmentation. Recently, Jing *et al*. [104] reconstruct images from events in the offline training phase of their unsupervised domain adaptation method to narrow down the gap between the source image domain and the target event domain and generate hybrid pseudo labels. Similarly, [105] and [106] employ event-based intensity image reconstruction to bridge the gap between image and event domains in their unsupervised domain adaptation frameworks.

It is also possible to use features produced within event-based video reconstruction networks, or the final reconstructions per se, as an intermediate feature for other tasks. In [107], Duan *et al*. employ an event-to-image module and make use of features learned at it as an intermediate feature for their event denoising and super-resolution network. Similarly, Ahmed *et al*. [108] employ an image reconstruction sub-network for event-based stereo disparity estimation and fuse learned image features with event features to aid in stereo matching. On the other hand, for the task of sleep activity recognition with event cameras, Plou *et al*. [109] employs intensity frames reconstructed from events as an additional input channel in their event representation to boost classification in challenging low-light scenarios.

As a final and distinct example, Ahmad *et al*. [110] use an event-based video reconstruction network to play the role of privacy attack and train their event anonymization network with a loss function designed to ensure that the reconstructed images do not contain person identity information. As demonstrated by these examples, event-based image reconstruction is a valuable component for other event-based vision tasks as well.

1.3. Scope of the Thesis

Following the motivations outlined above, this thesis focuses on the problem of event-based video reconstruction, that is, generating a sequence of intensity images from event streams. We limit our scope to generating images using only events as input, in contrast to the line of works that fuses event data with complementary lower-quality intensity frames to enhance them. Since the events that we use as input are mostly grayscale, the generated intensity

frames are grayscale as well, but this does not pose a limitation to our task: It is possible to generate color images (as we describe in Section 5.7. and show results in Section 7.3.) when events are acquired by a color event camera such as the ColorDAVIS346 [111].

We also constrain our task to the online scenario where the *future events* are not observed yet, and each generated image only depends on past events, in contrast to works that process event stream bidirectionally in an offline manner. Therefore, our task definition allows us to reconstruct intensity images from a continuous event camera stream in real-time. Following this, we also put a special emphasis on the computational efficiency of methods, in contrast to some of the recent works (*e.g.* [112, 113]) that only focus on the image quality aspect. Event cameras are known for their desirable properties, such as low-latency and non-redundant data flow, which make them ideal for scenarios that require real-time and low-power processing. Thus, we aim to pursue methods aligning with these scenarios while generating high-quality reconstructions.

Complementary to the pursuit of such methods, an equally important element is the evaluation of these methods. As we present in this thesis, reconstructing images from events is a complex task, depending on many variables that can affect the performance of the methods, not to mention the inherently subjective nature of image quality assessment. These aspects make evaluation challenging, which is often overlooked in the newly emerging literature. Therefore, it is crucial to evaluate the methods' sensitivity under varying conditions and challenging scenarios in a unified pipeline for fair comparison. Another important requirement for evaluation is a diverse set of test datasets that cover various real-world settings. It is well known that large-scale benchmarks have been instrumental in advancing many frame-based computer vision tasks (*e.g.* [114, 115]). However, since event-based vision is a relatively new field compared to classical frame-based computer vision, the current datasets used for assessing event-based video reconstruction are limited in scale and scope, confined to specific domains, scenes, camera types, and motion patterns. Therefore, we also aim to evaluate, analyze, and compare event-based video reconstruction methods comprehensively using a large set of real-world datasets, scenarios, and settings.

Given our task definition and the aforementioned scope of our research, we aim to facilitate progress in this research field by contributing to three main aspects of event-based video reconstruction in this thesis: dataset, evaluation, and method. An overview of our contributions are presented in the following section.

1.4. Contributions

As mentioned above, we contribute to three main aspects of the event-based video reconstruction research: dataset, evaluation, and method. Then, we conduct a rigorous experimental evaluation of our method, using our evaluation framework, and utilizing our proposed dataset as well as existing ones. In this section, we give an overview of our contributions presented in this thesis.

First, in Chapter 4., we introduce a new event dataset named HUE, for assessing the quality of reconstructed images. When compared to existing benchmark datasets, this dataset has a higher resolution and contains a larger number of sequences taken in diverse scenarios. Furthermore, a significant part of the proposed dataset specifically focuses on low-light scenarios, a challenging but rewarding domain for event-based video reconstruction.

Second, in Chapter 5., we turn our focus to the evaluation of event-based video reconstruction methods. Specifically, we identify several challenges and limitations in existing works and propose an open-source evaluation framework named EVREAL, which addresses these issues. Specifically, EVREAL provides a unified evaluation methodology to benchmark and analyze event-based video reconstruction methods from the literature. Our benchmark includes additional datasets, metrics, and analysis settings that have not been reported before; such as challenging scenarios involving rapid motion, low light, and high dynamic range. EVREAL also allows us to analyze the robustness of methods under varying settings such as event rate, event tensor sparsity, reconstruction rate, and temporal irregularity. Furthermore, EVREAL provides quantitative analysis on three downstream tasks (camera calibration, image classification, and object detection), enabling us to assess the performance of each method considering a specific downstream goal.

Third, we present HyperE2VID in Chapter 6., our proposed event-based video reconstruction method. In contrast to the existing works from the literature, which process the highly varying event data with static networks, HyperE2VID employs a novel dynamic neural network architecture with hypernetworks, per-pixel dynamic convolutions, and a context fusion block. Our experiments reveal that this dynamic architecture can generate higher-quality videos than previous state-of-the-art while also reducing memory consumption and inference time.

Finally, in Chapter 7., we evaluate HyperE2VID via extensive experiments and compare it with existing methods. Using EVREAL, we perform evaluation on several datasets, including our proposed one. We utilize several full-reference and no-reference image quality metrics to assess the performances of each method under various conditions. EVREAL also provides us with results on camera calibration, image classification, and object detection, allowing us to measure reconstruction quality via these downstream tasks. Furthermore, we analyze the robustness of each method concerning several variables of event-based video reconstruction tasks, such as event rate, event tensor sparsity, reconstruction rate, and temporal irregularity. We also analyze the computational complexity of each method. Finally, we present a detailed ablation study to validate many of the design choices of HyperE2VID.

The main contributions of this thesis can be summarized as follows:

- We collect and share a new event dataset with high resolution, a large number of sequences, diverse scenarios, and a specific focus on challenging cases like low-light scenarios.
- We propose a unified evaluation methodology and an open-source framework called EVREAL to benchmark and analyze event-based video reconstruction methods from the literature. We also present an online and interactive results analysis tool to visualize and compare reconstructions and their scores.

- Our benchmark includes a large set of datasets, metrics, and analysis settings, many of which have not been reported before. In particular, we present quantitative results on challenging scenarios involving rapid motion, low light, and high dynamic range. Moreover, we analyze the robustness of methods under varying settings such as event rate, event tensor sparsity, reconstruction rate, and temporal irregularity.

- To further examine the quality of the reconstructions, EVREAL provides quantitative analysis on three downstream tasks: camera calibration, image classification, and object detection. This extrinsic evaluation can be considered a proxy metric for image quality or a task-specific metric if event-based video reconstruction aims to perform these downstream tasks.

- We propose the first dynamic network architecture for video reconstruction from events, HyperE2VID, where we extend existing static architectures with hypernetworks, dynamic convolutional layers, and a context fusion block. Our per-pixel dynamic convolutions excel in adapting sparse and varying event data, while the context fusion module leverages complementary elements of event and frame domains. We employ filter decomposition for per-pixel filters, allowing us to reduce the computational cost significantly. We also present a curriculum learning strategy specific to our task and network, enabling robust training.

- We show via extensive experiments that this dynamic architecture can generate higher-quality videos than previous state-of-the-art while also reducing memory consumption and inference time.

1.4.1. Publications

Parts of these contributions are published in the following papers:

- Burak Ercan, Onur Eker, Aykut Erdem, and Erkut Erdem. EVREAL: Towards a comprehensive benchmark and analysis suite for event-based video reconstruction.

In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pages 3942–3951. 2023.

- Burak Ercan, Onur Eker, Canberk Saglam, Aykut Erdem, and Erkut Erdem. HyperE2VID: Improving event-based video reconstruction via hypernetworks. IEEE Transactions on Image Processing, 33:1826–1837, 2024.

1.4.2. Software Contributions

We also contribute the following software by open-sourcing them:

- The code for EVREAL (Event-based Video Reconstruction Evaluation and Analysis Library) is provided at `https://github.com/ercanburak/EVREAL`
- The interactive results analysis tool complementing EVREAL can be found at `https://ercanburak-evreal.hf.space/`
- The code for HyperE2VID is shared at `https://github.com/ercanburak/HyperE2VID`

1.5. Thesis Organization

We now present the organization of the thesis by providing an overview of the chapters and their contents:

Chapter 1 presents a brief introduction to event cameras, our task definition and motivation, our contributions, and the scope of the thesis. To put the scope of this thesis into a larger context, Chapter 2 gives a more detailed view of the event-based vision literature by presenting various types of event cameras and their advantages, a description of the event generation mechanism, various event representations and processing methods proposed in the literature, and vision tasks and applications targeted with these methods.

In Chapter 3, we review the literature on event-based video reconstruction by illustrating methods, evaluation procedures, and benchmark datasets. We discuss their limitations and motivate our contributions.

Chapter 4 details our proposed dataset, HUE. We begin by providing the motivation for introducing a new dataset, emphasizing key aspects that distinguish it from existing ones. Next, we offer a detailed description of the setup used to collect our dataset, including the cameras, lenses, configurations, and software employed. Finally, we describe the dataset itself by specifying the recorded scenes, listing sequences and their statistics, and providing sample frames and event visualizations.

Chapter 5 introduces EVREAL, our proposed open-source library for evaluating and analyzing event-based video reconstruction methods. Here, we provide a formal task description and explain our framework, including evaluation via image quality metrics and downstream tasks. We also describe the event-based video reconstruction methods we compare with EVREAL and the datasets and metrics used.

In Chapter 6, we present our proposed method, HyperE2VID. We describe our hypernetwork-based novel dynamic neural network architecture and training details, which improve the current state-of-the-art methods in terms of both image quality and efficiency. Here, we underscore the pivotal aspects of our work, such as per-pixel dynamic convolutions that adapt to sparse and varying event data, context fusion that leverages complementary elements of event and frame domains, filter decomposition steps for reduced computational cost, and curriculum learning for robust training.

Chapter 7 presents our experimental work by describing our setup, providing results of our extensive experimental work, and discussing them. Here, we evaluate HyperE2VID and other compared methods via several datasets, full-reference and no-reference image quality metrics, and downstream tasks. We also provide analyses on model robustness and computational complexity and present a detailed ablation study to assess the impact of various design elements of HyperE2VID.

Chapter 8 concludes the thesis by summarizing our contributions and discussing the obtained results. We also analyze the limitations of our work and provide possible directions for future research.

2. BACKGROUND

In this chapter, we delve deeper into the event-based vision literature by discussing various types of event cameras and their benefits, describing the mechanism of event generation, reviewing different event representations and processing methods found in the literature, and examining the vision tasks and applications addressed by these methods. Figure 2.1 displays pictures of example event cameras.

Figure 2.1 Pictures of example event cameras. From left to right: DVS128, miniDVS, and DAVIS346 by iniVation, EVK4 from PROPHESEE, and CeleX-V from CelePixel Technology.

2.1. Event Cameras

2.1.1. Types of Event Cameras and Bio-Inspired Vision Sensors

There are different types of event-based cameras according to their pixel structure and output they produce:

- **Dynamic Vision Sensor (DVS)** [69]: DVS outputs intensity change events per pixel; with pixel location, polarity, and timestamp. Most of the time, the terms *event camera* and *event data* are used to refer to the DVS type of cameras and data produced by them, respectively.

- **Asynchronous Time-based Image Sensor (ATIS)** [70]: ATIS produces two different types of events. When the intensity of a pixel changes, it outputs not only a *change event* similar to DVS but also *grayscale encoding events* which encodes the instantaneous intensity level of that pixel. This encoding is done via the time difference between two consecutive grayscale encoding events.

- **Dynamic Pixel and Active Vision Sensor (DAVIS)** [71]: DAVIS combines a DVS with a standard frame-based sensor in the same pixel array. It outputs intensity frames with a specific frequency like a standard camera, in addition to asynchronous events like DVS. It is also possible to include a color filter array or a polarizer array in front of DAVIS pixels, to build prototypical event cameras that are capable of outputting color or polarization events, such as ColorDAVIS346 [111] and PDAVIS [116], respectively.

- **VidarOne Spike Camera** [117]: The pixels of a spike camera continuously accumulate light and fire a spike signal when the accumulated intensity exceeds a threshold. Unlike DVS, which only captures changes in relative light intensity, a spike camera can record the absolute light intensity, encoded as the frequency of spike firing. This allows it to *see* both static scenes and scenes with fast motion, with the expense of increased and redundant data flow.

In this thesis, we focus on brightness change events generated by a DVS-like event camera and use the term *event* to refer to these brightness change events.

2.1.2. Event Generation

Here, we more formally define the event generation mechanism by first presenting a simple model of the event camera pixels. Then, by highlighting some of the unrealistic aspects of this simple model, we will guide the discussion toward a more realistic and complex model.

Simple Event Generation Model: Let $I(x,t)$ denote the light intensity level over pixel location x at time t, and $L(x,t) \doteq log(I(x,t))$ denote the log-intensity level. Each pixel of an event camera inherently stores a reference log-intensity level $L(x,t_r)$ where t_r is the time of the last generated event at that pixel. They continue to monitor the log-intensity $L(x,t)$ with high temporal resolution, and when the difference between current and reference log-intensity values reaches a positive constant C called *contrast threshold* such that Equation (1) holds, they instantly generate an event $e \doteq (x,t,p)$, where polarity $p \in \{+1,-1\}$ is the sign of the brightness change.

$$L(x,t) = L(x,t_r) + pC \tag{1}$$

When the event is generated, the reference log-intensity value of the pixel is reset to its new value, and the generated event signal is asynchronously output from the event camera independent of other pixels. The event signal carries information about the pixel location x, the timing of the event with the precise timestamp t, and polarity p of the event.

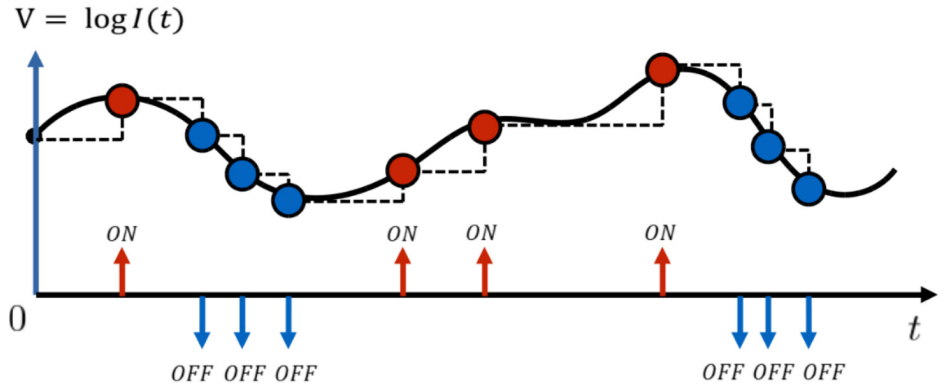


Figure 2.2 A visualization of the simple event generation model. For a single pixel, ON and OFF events are generated asynchronously and sparsely in time (horizontal axis), according to the variations of the log-intensity values of that pixel (vertical axis). Figure taken from [11].

A More Realistic Approach: Equation (1) describes a simplistic model of event generation. In reality, the contrast threshold C is not constant but can vary across pixels, between positive and negative events, and with changes in illumination, temperature, sensor noise, and sensor parameters. Moreover, the simple model we present above does not account for various effects in the event pixel, such as the refractory period (or dead time, duration for which the pixel is blind after each event), noise [118], or junction leakage events [119]. Therefore, a more realistic approach is to model the contrast threshold C with a probability distribution as in [97], or using an event generation model that accounts for inter-pixel bias and contrast threshold variations as in [120] or [121].

Photometric Constancy Equation: For a pixel, let Δt be the elapsed time and ΔL be the change of log-intensity after the last event. Assuming small Δt, constant illumination, and

Lambertian surfaces, one can show as in [100] that ΔL can be approximated by the below equation:

$$\Delta L \approx -\nabla L \cdot \mathbf{u} \Delta t \tag{2}$$

where ∇L is the spatial gradient moving on the image frame with optical flow vector $\mathbf{u}$. This equation is called the event-based photometric constancy equation, and it describes the fact that the log-intensity value of a pixel changes due to the moving edges in the scene. When the movement is parallel to the intensity edge, the log-intensity value of the pixel does not change, and no event is generated. When the movement is perpendicular to the intensity edge, the rate of change of the log-intensity value is highest, and therefore, events are generated at the highest rate.

2.1.3. Advantages of Event Cameras

Thanks to the unique operating principles and aforementioned event generation mechanism, event cameras possess many advantages compared to traditional frame-based cameras. These advantages can be summarized as follows:

- **High Dynamic Range:** Event cameras have a very high dynamic range ($140\,\mathrm{dB}$) compared to that of regular frame-based cameras ($60\,\mathrm{dB}$) [74]. Therefore, they do not lose information like them due to over or under-exposed parts when there are strong lighting variations within a scene.
- **High Temporal Resolution:** Each pixel of an event camera timestamps events with a $1\,\mathrm{MHz}$ clock, resulting in a microsecond temporal resolution. This pixel-wise high temporal resolution brings an advantage that was not possible with conventional frame cameras. In [122], authors argue that this brings 70% more information for pattern recognition tasks, thus drastically increasing separability between classes of objects.

- **Low Latency:** Since each pixel generates events for local log-intensity changes without waiting for global synchronization, the changes in the scene are transmitted with very low latency, in the order of tens to hundreds of microseconds. Therefore, real-time interaction and control systems can respond to changes in the visual scene very fast.

- **Low Power:** Unlike frame-based acquisition, event cameras transmit non-redundant events, which are generated only when pixel-wise intensity changes. Therefore, power needs to be used only for these non-redundant processing and transmission, resulting in a low overall power consumption, which is highly desirable for applications demanding low power consumption such as mobile robotics, wearable electronics, IoT, and AR/VR.

- **Others:** Event cameras have other advantages, some of which are related to or overlapping with the above ones, such as minimal motion blur, the low bandwidth requirement for transmission, and low memory consumption for visual data.

2.2. Event-based Vision

As mentioned in Section 1.1., the advantages of event cameras and their increasing availability have led to a natural interest in utilizing them for visual tasks that are difficult with standard cameras. However, achieving this goal also presents some challenges since directly employing successful methodologies from frame-based computer vision is impractical due to the distinct nature of data generated by event-based sensors.

Output from an event camera is a temporally continuous stream of asynchronous events, and each event conveys very little information regarding the scene. Processing methods need to account for this continuity and find ways to extract meaningful information from events. Therefore, researchers have come up with many different event representations and methods for processing them to adopt event-based data in different applications. As we present in this chapter, there are distinct approaches proposed in the literature on how to manipulate, represent, and process events, with each approach having its own advantages

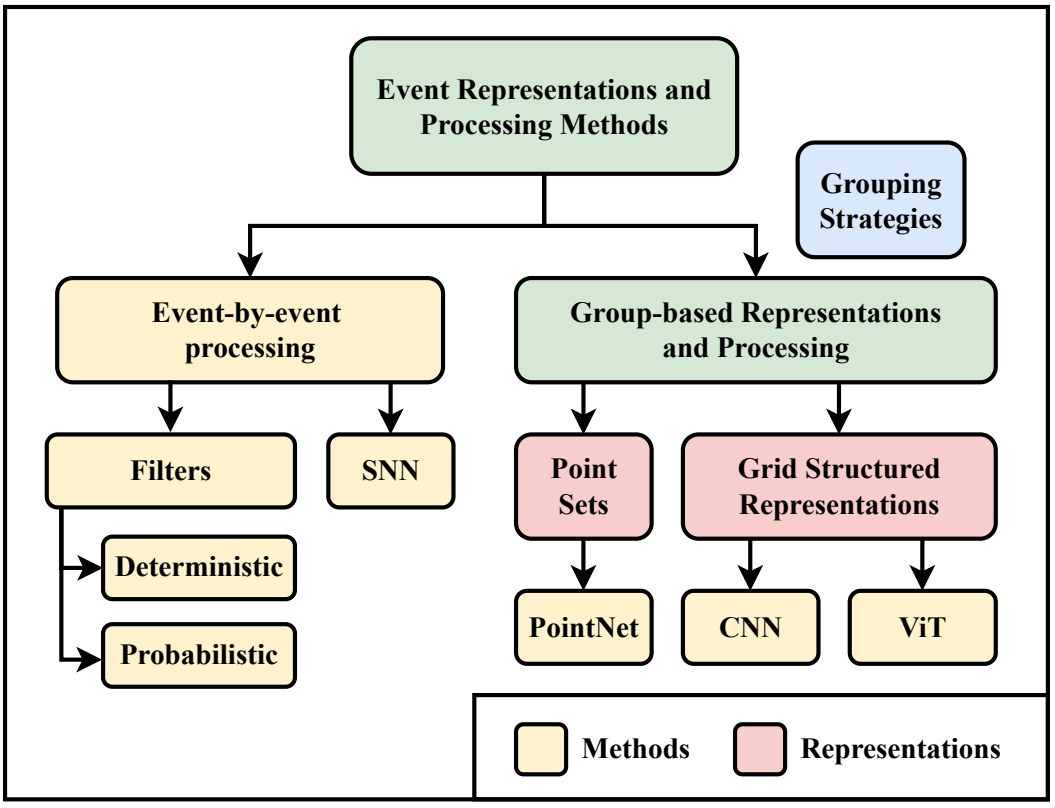


Figure 2.3 A diagram showing notable event representations and processing methods.

and disadvantages. Some notable representations and methods are summarized in Figure 2.3, which can give a general idea before delving into the more comprehensive discussion that follows.

2.2.1. Event Representations

One way to process events is taking them **one by one** and processing each of them separately. This can be done with filter based approaches (*e.g.* [123, 124]) or spiking neural networks (SNNs) (*e.g.* [125, 126]). In these methods, each event acts by changing the internal state of the processing model to yield desired outputs. The second approach is to aggregate information from a **group of events**, for example, by considering a spatiotemporal

neighborhood and then processing this group together. Researchers have proposed several **event representations** based on event grouping, which we cover below.

One option is to obtain a group-based representation is accumulating events onto an **image-like 2D grid**. This can be done by aggregating events for each pixel by summing their polarities ($+1$ for positive events and -1 for negatives) [127], accumulating events with different polarities in separate image channels [128], considering polarity of the latest event for each pixel to obtain a ternary image [129], or computing pixel-wise event count histograms [130]. A similar 2D representation is **time surface**, where the timestamp of the latest event determines the value for each pixel [131]. Instead of employing the timestamp of the latest event, the average timestamp of events from each pixel can be used to create a **time image** [132].

Another type of group-based representation considers the events as a **3D point set** [133]. This representation is similar to point clouds, with one dimension of the 3D space being temporal. Just like the voxelization of point clouds, event point sets can also be converted to a **3D voxel grid** representation [99, 134]. Another similar representation considers events as **evolving 2D point sets** on the image plane [135, 136]. There are also representations like TORE [137] and MDOE [138], which are **4D volumes** focusing on retaining timestamp and polarity information instead of aggregating them.

With event-by-event processing, it can be possible to obtain a low latency output, since there is no need to wait for a certain number of events to accumulate. However, it may require more computation since every event is processed separately. On the contrary, group-based representations introduce more latency but may require less computation after the pre-processing is done for the representation.

Some group-based representations like point clouds may be more informative and representative than others, since they do not discard information by aggregating events, at the expense of more computation and memory requirements. Image-like representations often discard some information like polarity or precise timestamp by summing events with different polarities or by averaging or quantizing timestamps in favor of the easier

computation obtained. Time surfaces do not quantize timestamps of events and keep the high-resolution timing information. However, they still suffer from information loss when multiple events from the group overwrite to same pixel.

An advantage of image-like representations is their compatibility with conventional computer vision methods. However, this comes at the expense of discarding information that may be valuable for the task at hand. Representations like voxel grids also quantify timestamps, but they better preserve the space-time structure while still being compatible with conventional computer vision methods like CNNs. However, they lose the sparsity of the events and require more memory.

Since each event representation has its advantages and disadvantages, it is important to select the right representation considering the task, scene, assumptions, and other constraints like available computational resources. As an example, Jiao *et al*. [139] empirically compares different event representations for event-based SLAM. It is also possible to combine strong points of different representations for the task at hand. For example, the methods in [140] and [141] combine event-count-based representations and timestamp-based representations in separate channels of their image-like representations.

Motivated by the fact that optimal representation for a specific task may be hard to select and tune, the authors of [142] and [143] propose to learn a representation together with the task in an end-to-end manner, rather than using fixed representations. Wang *et al*. [144] argue that these two learned representations are primarily designed for object classification and might not be optimal for location-oriented tasks like object detection, and suggest a new learned representation that capture spatial-temporal information.

In contrast to these task-specific learned representations, the learned representation of [145] is task agnostic and thus can be used with various tasks once trained. Furthermore, the authors of [146] argue that the learned representation of [143] is not suitable for image enhancement tasks due to the loss of connection around neighboring pixels and propose to learn representations via bidirectional ConvLSTM blocks. However, this method encodes the

event stream bi-directionally in an offline manner and thus unsuitable for online prediction tasks.

Rather than training a new representation, Zubić *et al*. [147] parameterize a family of event representations based on a stack of simple features, and select a representation by optimizing the hyperparameters. The optimization is performed by measuring the discrepancy between events and their representation according to the Gromov-Wasserstein Discrepancy (GWD) on the validation set.

2.2.2. Event Grouping Strategies

An important consideration in group-based representations is determining the event grouping (or selection) strategy. Three prominent strategies are as follows:

Fixed-number: Grouping every N_G number of events such that the kth event group can be defined as:

$$G_k \doteq \{e_i \mid kN_G \leq i < (k+1)N_G\} \tag{3}$$

Here, the rate at which the groups are formed varies according to the incoming event rate.

Fixed-duration: Grouping events according to non-overlapping time windows with a fixed duration of T_G secs. The kth event group contains all events with timestamps t_i falling within the kth time window, defined as:

$$G_k \doteq \{e_i \mid kT_G \leq t_i < (k+1)T_G\} \tag{4}$$

In this scheme, the number of events in each group varies according to the incoming event rate.

Between-frames: Assuming that the ground truth intensity frames are available together with the incoming event stream, we can group events such that every event between

consecutive frames belongs to the same group. Therefore, the set of events in the kth event group can be defined as follows:

$$G_k \doteq \{e_i \mid s_k \leq t_i < s_{k+1}\} \tag{5}$$

If the ground truth frames arrive at a fixed rate, then this option is a special case of the fixed temporal window grouping. Note that, however, the time difference between consecutive frames may not be fixed all the time, due to changing camera exposure times in real-world datasets or due to adaptive rendering schemes of simulators in synthetic datasets.

For the **fixed-number** and **fixed-duration** strategies, an important consideration is determining the number of events or time duration, respectively. Furthermore, these event selection windows can be made overlapping as well instead of being non-overlapping. In the extreme case, one can use a **sliding window approach**, and update the representation by each event. Each of these design choices comes with their trade-offs.

The fixed-duration strategy discards the asynchronous nature of the event stream by generating a representation with a constant frequency, regardless of the motion in the scene. Therefore, it can result in groups with too few or too many events. Representation with too few events may convey inadequate information for the task at hand. On the other hand, having too many events may mean information loss due to the aggregation schemes of representations.

The fixed-number strategy better respects the asynchronous nature of the event stream, but results in temporally irregular event groups. This might not be desirable for downstream tasks, and therefore tuning the number of events in this strategy may be difficult. By using overlapping event windows, it can be possible to generate an output more frequently and obtain a lower latency at the cost of more computational requirements.

Whether using the fixed-duration or fixed-number strategy, scene characteristics can have a profound effect on downstream task performance. For densely textured scenes, the same

amount of motion creates more events per time interval, when compared to scenes with sparse texture. Thus, selecting a fixed event number or time duration may be sub-optimal for particular scenes. To alleviate this problem, it is also possible to use an adaptive event selection strategy [8, 148–151].

2.2.3. Event Processing Methods

Methods of event processing are closely related to the representations they use since the structure of the representation generally determines certain classes of methods that can be applicable. Furthermore, the boundary between event representations and methods for processing them are not well defined. For example, the method in [152] enhances time surface representation with a memory structure that enables incorporating the information carried by past events, to obtain a higher-order representation that is more robust to noise or small variations in the event stream; while Deng *et al*. [153] use a deep residual network to learn the parameters of their adaptive motion-agnostic event encoder.

2.2.3.1. Event-by-event Processing We have already stated that filter-based approaches or spiking neural networks are typical with event-by-event processing. Filter-based approaches include **deterministic filters** such as motion-direction sensitive spatio-temporal filters for optical flow detection [154], and per-pixel temporal high-pass filter for image reconstruction [90]; as well as **probabilistic filters** such as particle filter to estimate the rotation motion of camera and pixel-wise extended Kalman filter (EKF) to estimate scene gradients [97].

Spiking neural networks seem like a natural choice for processing event data due to their bio-inspired and asynchronous processing. However, they are hard to train since spike signals are not naturally differentiable like layers of standard artificial neural networks (ANNs). Although there are works that propose differentiable synapse models or formulations that approximate backpropagation for spiking neurons [155–158], these methods are not as popular as their ANN counterparts, with few applications [159]. The authors of [160]

propose a method that unrolls the SNN in time at high resolution to train them, but this brings heavy computation and memory requirements. Another approach is to train an ANN first and then convert the trained network to SNN [161–165]; but the obtained network performance highly depends on a set of parameters like refractory time and leakage rate, and may not reach the performance of the original network even after tuning these parameters. Another reason why SNN-based methods are not as popular as ANN-based ones is the fact that they require specialized neuromorphic hardware [166, 167] to perform efficient inference with them. There are also works that employ ANNs and SNNs in conjunction, to combine their advantages [168, 169].

Other event-by-event processing methods in the literature are **asynchronous convolutions** [170–173], graph-based asynchronous event processing [174], and PointNet like event-by-event update of [133]. A distinct approach to process asynchronous event data one by one is to use a **Neural Ordinary Differential Equation (NODE)** [175] based method, as presented in [176].

2.2.3.2. Group-based Processing For group-based representations that have a certain grid structure (like 2D image-like grids including time surfaces, or 3D voxel grids), a natural choice available is the use of modern DNNs. Although some works used methods that employ **hand-crafted feature extraction** [177–180]; nowadays it is more common to see **CNN** based methods applied on a grid structured representations [128, 134, 140, 181–183].

Other than CNNs, researchers also proposed methods that make use of other prominent classes of deep neural networks such as **recurrent neural networks (RNNs)** [87, 184, 185], **generative adversarial networks (GANs)** [78, 186, 187], **graph convolutional networks (GCNs)** [188–190], transformers [151, 191–195], diffusion models [113, 196], and state space models (SSMs) [197, 198].

2.2.4. Tasks and Applications

Here, we briefly present various vision tasks and applications attempted by event data by giving pointers to representative works from respective areas. We purposefully omit the intensity reconstruction task here, as we cover it in more detail in Chapter 3. The list of event-based vision tasks can be seen in Table 2.1, while some application areas are presented in Table 2.2.

Table 2.1 List of some event-based vision tasks with pointers to respective works.

Object Recognition	[164] [188] [199] [180] [176] [182] [200] [174] [194]
Optical Flow	[140] [201] [202] [203] [204] [205] [206] [207] [208]
Visual (Inertial) Odometry	[209] [210] [211] [212] [213] [214] [215] [216] [217]
Object Tracking	[218] [219] [220] [221] [222] [223] [224] [225]
Feature Detection & Tracking	[136] [226] [178] [9] [179] [227] [228] [229]
Stereo Depth Estimation	[230] [231] [232] [233] [234] [235] [236]
Motion Segmentation	[237] [238] [189] [239] [240] [241]
Semantic Segmentation	[242] [103] [243] [244]

Table 2.2 List of some event-based vision application areas with pointers to respective works.

Robotics	[245] [246] [247] [248] [249] [250]
Driver Monitoring	[251] [252] [253] [254]
Space	[255] [256] [257]
Eye and Gaze Tracking	[258] [259] [260]
Fall Detection	[261] [262] [263]
Lip Reading	[264] [265] [266]
Microscopy	[267] [268] [88]
Visual Inspection	[269] [270]
Animal Behavior Monitoring	[271] [272]
Biometric Authentication	[273]

These are not meant to be comprehensive lists, and our aim here is to give the reader an idea about the scope of the event-based vision field. A more comprehensive survey can be found in [74]. Other survey/review papers focus on some specific tasks or domains in the field. The review paper of Lakshmi *et al*. [274] focuses on the tasks of object detection/recognition,

object tracking, and localization and mapping. The survey of Steffen *et al*. [275] concentrates on event-based stereo vision. The study of Chen *et al*. [276] focuses on event-based vision applications for autonomous driving. In two recent studies, Zheng *et al*. [75] focus on deep-learning techniques for event-based vision, while Huang [277] review event-based visual simultaneous localization and mapping (SLAM). Finally, the survey in [278] discusses automotive applications of event cameras for in-cabin and out-of-cabin monitoring.

3. RELATED WORK

As presented in the previous chapter, event data has been increasingly incorporated into various tasks due to its numerous advantages (see Section 2.1.3.). These include recognition tasks such as object detection [149], semantic segmentation [279], and fall detection [263]. Furthermore, event data has been utilized in challenging robotic applications that require high-speed perception, such as an object-catching quadrupedal robot [250] and an ornithopter robot capable of avoiding dynamic obstacles [248].

However, event-based data is an entirely different visual modality compared to standard intensity frames. While it has many desirable properties, we can not directly interpret event streams as we do for intensity images. Hence, reconstructing intensity information from event data has long been a cornerstone in event-based vision literature, with earliest works like [96] being nearly as old as the first line of prototypical event cameras. Since then, there have been many works on the subject, including some impressive results in recent years (*e.g.* [12]). Nevertheless, we consider the problem as still being far from solved, considering the fact that state-of-the-art approaches use event representations that cause latency, rely on carefully designed large synthetic datasets, use computationally expensive models, and still produce reconstructions that suffer from some complications, such as unrealistic artifacts.

As we present in this chapter, there are distinct approaches to this problem in the literature, with different assumptions, constraints, and processing methodologies. We first review these methods in detail in Section 3.1. Next, we present evaluation methodologies for this task in Section 3.2. Finally, we provide benchmark datasets for the intensity reconstruction task in Section 3.3.

3.1. Methods

The event generation model given in Equation (1) describes the fundamental relation between intensity information and generated events. Since each event encodes an intensity change information, one can accumulate events of a pixel x for a time interval according to their

polarity to approximate the total intensity change for that pixel. Then, if the absolute intensity value at the start of that time interval is known, one can obtain the final approximate intensity value of that pixel. This *direct integration* procedure can be formalized by the equation below as shown in [280]:

$$\log \hat{I}(x,t) = \log I(x,t_0) + \sum_{t_0 < t_i \leq t} p_i \, C \, \delta_k(x - x_i)\delta_d(t - t_i) \tag{6}$$

where t_0 is the start of the time interval, δ_k is the Kronecker delta function (with discrete variables), and δ_d is the Dirac delta function (with continuous variables).

However, there are several limitations to this simple direct integration approach. First, the initial intensity value $I(x, t_0)$ may be unknown. Second, integrating events can only approximate the actual intensity value due to sampling and quantization, *i.e*. one can not infer the intensity variations smaller than C between consecutive events. And most importantly, the Equation (6) relies on the simplistic event generation model of Equation (1), ignoring the true stochastic nature of events [118, 281, 282].

Due to the problems associated with the direct integration approach, the intensity reconstruction task requires incorporating additional information or assumptions. Earlier works on intensity reconstruction (Section 3.1.1.) generally rely on some limiting assumptions such as known or restricted camera movement, static scenes, or brightness constancy. On the other hand, recent deep learning based methods (Section 3.1.2.) incorporate natural image priors in their models through the learning process. However, this learning process is generally made possible via large synthetic datasets, which need to be carefully designed, and brings the question of generalizability to real-world scenarios. We present a high-level overview and categorization of these methods in Table 3.1.

Pioneering works, initiated by Cook *et al*. [96], typically aimed at simultaneously estimating multiple quantities like intensity images, spatial gradients, and optical flow [97–99]. This multi-faceted approach benefits from the dynamic interaction between these elements, as exemplified by the event generation model of Gallego *et al*. [100], which correlates optical

Table 3.1 Categorization of intensity reconstruction works from the literature. An asterisk symbol (*) at rightmost column indicates the availability of open-source implementation for the given work.

Filter Based	Probabilistic Filters and Integration		[97]	*
			[98]	
	Temporal Filter and Integration		[283]	
			[90]	*
Optimization Based	Message Passing		[96]	
	Variational Optimization		[99]	
			[284]	*
	Linear Inverse Problem		[285]	*
Learning Based	Unsupervised	Sparse Dictionary	[286]	
		GAN	[187]	
			[85]	
			[78]	
			[287]	
			[288]	
	Self-Supervised	RNN	[101]	*
	Supervised		[12]	*
			[91]	*
			[289]	*
			[290]	*
			[291]	*
			[112]	*
		SNN	[292]	*

flow, scene gradients, and event data. In methods that primarily predict scene gradients, a common subsequent step involves employing Poisson integration [293] to derive intensity images from these gradients.

Kim *et al*. [97] introduced a filter-based method for estimating scene gradients and ego-motion, but it was limited to rotational camera movements. They later expanded this work in [98] to accommodate free camera motion, though still confined to static scenes. Bardow *et al*. [99] approached dynamic scenes by variational optimization, estimating intensity images and optical flow under the brightness constancy assumption.

Barua *et al*. [286] were the first to show that motion estimation is not necessary for intensity

image reconstruction, employing a patch-based dictionary learning method. Following a similar vein, Munda *et al*. [284] proposed an optimization-based method, minimizing an energy function with a data fidelity term based on direct event integration and a manifold regularization term. Scheerlinck *et al*. [90] also used event integration but added a per-pixel temporal high-pass filter to mitigate noise. While their approach allowed for continuous time processing of events, it resulted in artifacts due to the loss of low-frequency information from static backgrounds.

The last few years have witnessed many works that utilize neural networks and deep learning methodologies for the task of intensity image reconstruction. Wang *et al*. [187] represented groups of events with spatio-temporal voxel grids and fed them to a conditional GAN to output intensity images. In their seminal work, Rebecq *et al*. [87] proposed a recurrent fully convolutional network called E2VID to which they input voxel grids of events to produce an intensity image. They trained this network on a large synthetic dataset generated with ESIM [294] using the perceptual loss of [2] and showed that this generalizes well to real event data at test time. As a follow-up study [12], the authors employed temporal consistency loss [3] to minimize temporal artifacts.

After E2VID, many works attempted to enhance it from various perspectives. Scheerlinck *et al*. [91] replaced E2VID architecture with a lightweight recurrent network called FireNet, which has much less memory consumption and faster inference. However, the reconstructions of FireNet were not as good, particularly in scenarios with fast motion. Stoffregen *et al*. [289] improved the results of E2VID and FireNet by matching statistics of synthetic training data to that of real-world test data, resulting in E2VID+ and FireNet+. Cadena *et al*. [291] employed spatially-adaptive denormalization (SPADE) [295] layers in E2VID architecture, improving the quality of reconstructed videos, especially for early frames, but with an increased computational cost. Similarly, Weng *et al*. [112] incorporated a Transformer [296] based module to the CNN-based encoder-decoder architecture of E2VID, improving the reconstruction quality at the expense of increased computational complexity.

In contrast to these, a few recent works followed somewhat different approaches, mainly

targeting aspects other than the quality of reconstructions. As an example, Paredes-Vallés and de Croon [101] turned back to the idea of simultaneously estimating optical flow and intensity images via photometric constancy assumption, and suggested a method based on self-supervised learning, eliminating the need for synthetic training data with ground truth frames. Zhu *et al*. [292] used a deep spiking neural network (SNN) architecture, targeting computationally efficient neuromorphic hardware. Zhang *et al*. [285] formulated the event-based image reconstruction task as a linear inverse problem based on optical flow, and suggested a method without training deep neural networks. Although these methods brought improvements in aspects like required training data, computational efficiency, or explainability, the visual quality of their reconstructions was not as strong.

There are also works that target a slightly different task. As an example, Zhang *et al*. [85] argued that the reconstruction performance of E2VID deteriorates when operated with low-light event data, and proposed a novel unsupervised domain adaptation network to generate intensity images as if captured in daylight, from event data of low-light scenes. Mostafavi *et al*. [290] presented a network to generate super-resolved intensity images from events. Similarly, Wang *et al*. [287] introduced a network that can also perform image restoration and super-resolution. The work of Zhu reconstruct images from spike camera data instead of DVS-like events, while the follow-up work in [297] propose a network to reconstruct video by using both DVS events and spike flow.

In the following, we delve deeper into intensity image reconstruction methods in the literature.

3.1.1. Earlier Approaches

One of the earliest works on intensity reconstruction is [96]. In this work the authors propose a bipartite-graph-like network similar to factor graphs, to simultaneously estimate multiple quantities. One set of nodes in bipartite graph are called maps, and correspond to quantities that are of interest: events, intensity image, spatial gradient, optical flow, camera rotation, and camera calibration. The other set of nodes are called relations or interactions and

correspond to the relations between these quantities. In this method, given source quantities, other quantities are updated continuously and simultaneously, trying to satisfy the relations between them, until convergence.

Kim *et al*. [97] propose a method for simultaneous mosaicing and tracking under a strong assumption on camera motion: only rotation. Their approach is similar to the methods of simultaneous localization and mapping, but since the only camera motion here is rotation, the constructed map of environment is not a 3D one, just a mosaic (panorama) image. They use a particle filter for ego-motion estimation, and a pixel-wise EKF to estimate scene gradients. After obtaining the spatial gradient map, they employ Poisson integration [131, 293] to reconstruct intensity images. Later, they extend this method to handle free 6-DOF motion, by employing three EKFs running in parallel to perform tracking, intensity reconstruction and mapping [98].

Nabil *et al*. [283] propose a 360° stereo panoramic camera system consisting of two 1D event cameras mounted on a mechanical device which continuously rotates around a single-axis with high-speed. Due to the constrained motion of cameras, they are able to generate 360° panoramic images by integrating events line-by-line. They use a temporal high-pass filter to decrease the low frequency noise components. Thanks to the stereo camera pair, they are also able to generate depth images.

Barua *et al*. [286] present one of the earliest works which do image reconstruction with an unsupervised learning approach, before deep learning based methods. They learn a patch-based sparse dictionary from simulation data, using K-SVD algorithm [298]. Then, this dictionary is used to obtain gradient images from events. Finally, they employ Poisson integration [131, 293] to reconstruct intensity images. This work also demonstrated that knowing or estimating camera motion is not required for intensity image reconstruction.

Bardow *et al*. [99] present a method where intensity image and optical flow field are simultaneously estimated. They solve a variational optimization problem that minimizes a cost function over a spatio-temporal volume of events, to obtain image reconstruction and optical flow. The cost function consists of flow and intensity smoothness terms, an optical

flow term to enforce brightness constancy assumption, and two more terms to account for events that were fired (or not). This method demonstrate intensity reconstruction without strong assumptions about camera motion, the authors claim that it works "while the camera undergoes a generic motion through any scene". They only make the brightness constancy assumption, *i.e*. events are only fired due to optical flow.

Similarly, Munda *et al*. [284] propose an optimization based method, using a framework of variational denoising. They consider a spatio-temporal volume of events and a lower dimensional manifold in this volume, defined by the timestamp of the latest event for each pixel, like time surfaces. They minimise an energy function consisting of a data fidelity term based on direct integration of events, and a manifold regularisation term. Similar to [286], their method does not need to estimate motion to do image reconstruction.

Method of Scheerlinck *et al*. [90] also builds on the idea of direct integration. They employ a per-pixel temporal high-pass filter before integration, to diminish accumulated noise. Unlike other works that use spatio-temporal filtering, they only use this temporal filter. Their method does not depend on a motion-model, and event-by-event processing enables intensity reconstruction in continuous-time: the internal state of the filter is updated asynchronously by each event, and one can query intensity information for any chosen time. However, due to high-pass filtering, low frequency information coming from static backgrounds are also lost in the process, and obtained images suffer from artifacts. Therefore, they propose to use a complementary filter structure to fuse events with grayscale frames from DAVIS when available, to obtain better images.

3.1.2. Deep Learning Era

Recent few years have witnessed many works that utilize neural networks and deep learning methodologies for the task of intensity image reconstruction, as we cover below.

Wang *et al*. [187] propose to use conditional GANs for the image reconstruction task. They stack events based on constant-time-duration and constant-event-number strategies. The

representation of event stacks is a voxel-grid structure. This representation is fed to the generator and discriminator networks of their framework, similar to pix2pix [299]. After training, the generator can output realistic intensity images, conditioned on the incoming events.

Rebecq *et al.* [87] employ a novel recurrent fully convolutional network based on U-Net [300] architecture. They split the event stream into non-overlapping windows with constant event numbers, and use a spatio-temporal voxel grid representation for event groups in each window, where each event distributes its polarity to the two closest spatio-temporal voxels, similar to bilinear interpolation. Voxel grid of each window is input to the RNN for each time step, and the output is an intensity reconstruction image generated from that time step, as illustrated in Figure 3.1. The overall neural network, which generates videos out of events, is called E2VID.

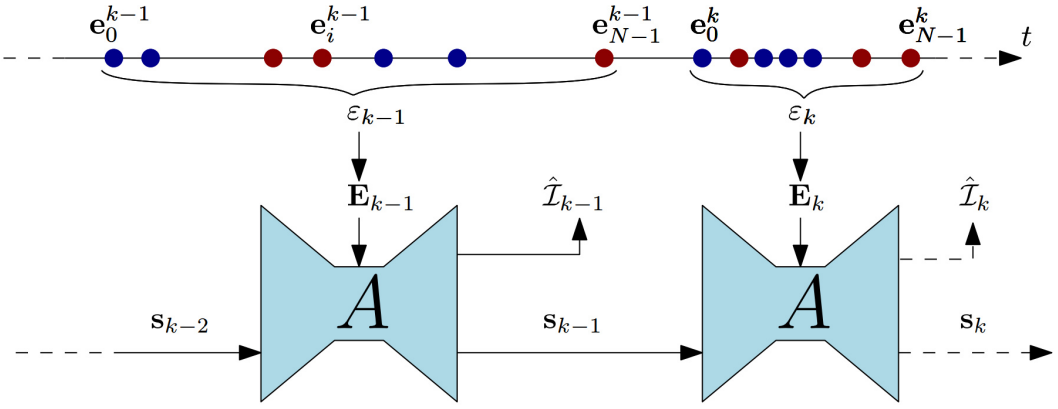


Figure 3.1 RNN based approach of Rebecq et al. Figure taken from [12].

E2VID is trained in a supervised manner, using a large synthetic dataset of event sequences with corresponding ground-truth intensity images generated in a simulation environment [294], with perceptual loss of [2]. Through experiments, they show that this training generalizes well to real event data at test time. Generated videos have high FPS and preserve HDR of event data. Furthermore, they also show that these reconstructions are good enough so that off-the-shelf computer vision algorithms can be used on them to perform downstream tasks of image classification, visual-inertial odometry, object detection, and monocular depth prediction with high performance, producing state-of-the-art results. In their follow-up work

[12], the authors employ temporal consistency loss [3] in addition to the perceptual loss, to minimize temporal artifacts like blinking. They also replace vanilla RNN layers with ConvLSTM [301] blocks, and train the network with longer event sequences.

Later, Scheerlinck *et al*. [91] propose a similar recurrent architecture called FireNet, which is much more lightweight compared to E2VID. They use GRUs instead of LSTMs, since they have less parameters while showing similar performance [302]. By also employing some other architectural changes; they are able to achieve much fewer network parameters (38K vs. 10M), faster inference time ($10\,\mathrm{ms}$ vs. $30\,\mathrm{ms}$ for GPU implementation at 640×480 resolution), 11.7 times fewer number of floating point operations, and much less memory consumption (0.16 Mb vs 43 Mb). With FireNet, they achieve comparable or better performance than E2VID on average. For challenging scenes that contain very fast motions, and during initialization; FireNet reconstructions exhibit some deficiencies compared to E2VID.

Stoffregen *et al*. [289] further improve results of E2VID [12] by carefully analyzing and matching statistics of synthetic training data to the statistics of existing real world data. For this, they create a new synthetic dataset by primarily changing the contrast threshold parameter of the simulation environment [294]. They also propose a method for dynamic train-time noise augmentation. Retraining of E2VID with these changes result in 20-40% improvement across multiple datasets and evaluation metrics.

Zhang *et al*. [85] argue and demonstrate that the reconstruction performance of state-of-the-art E2VID network [87] deteriorates when operated with low-light event data, since the synthetic training data may not model real noise distributions well due to the domain gap between daylight and low-light. They propose a novel unsupervised domain adaptation network to generate intensity images as if captured in daylight, from event data of low-light scenes. The idea behind their method is to extract domain invariant scene level features by separating them from domain-specific features similar to [303], and further enhance them by hallucinating plausible daylight-specific details that are missing in the low-light domain, in conditional generation manner. The method requires a dataset with events and corresponding

intensity images in daylight scenes, and a set of events in target low-light scenes without the need for corresponding intensity images.

Yu *et al*. [78] propose a network called Cycle-Event Network, which is an enhanced version of CycleGAN [304] where paired data is not required for the training phase. Furthermore, they propose a novel attention mechanism and employ it in the generator architecture, to diminish noise in backgrounds and enhance texture details in the reconstructed images.

Mostafavi *et al*. [290] present an end-to-end network to generate super-resolved intensity images from event-based data. They use a stacked event representation based on fixed-number event grouping strategy. They calculate optical flow from consecutive stacks with a pretrained off-the-shelf optical flow estimation network [305]. Calculated optical flow is input to an event feature rectification network together with the initial stacks, to obtain a fused event representation. Then, a recurrent super-resolution network is employed to generate intermediate super-resolved intensity images for each time step. Finally, three consecutive intermediate images are input to a mixer network, to obtain the final super-resolved intensity image with richer details.

Paredes-Vallés and de Croon [101] were the first to present a reconstruction method with a self-supervised learning approach. They argue that the previous methods which employ supervised training with large synthetic datasets suffer from simulation to reality gap. Their self-supervised approach builds upon several ideas proposed in the literature. First, they turn back to the event-based photometric constancy assumption [100] similar to some of the earlier reconstruction approaches (*e.g*. [99]), to simultaneously estimate optical flow and reconstruct intensity images. The architecture consists of an optical flow estimation network called FlowNet and a reconstruction network called ReconNet. FlowNet is trained via the contrast maximization proxy loss of [134], based on the contrast maximization framework presented in [306]. ReconNet is trained by making use of an image registration perspective [307], to reconstruct intensity images that best explain input events, assuming that the optical flow calculated by FlowNet is error-free. The optical flow network FlowNet and ReconNet can be instantiated as specific networks that are previously proposed in the literature, such

as EV-FlowNet [140] and E2VID [12]. Through evaluations, the authors show that their method can output promising results despite being a self-supervised approach. Common failure cases are due to inaccurate optical flow estimation, unknown initial intensity value, and large texture-less regions.

Cadena *et al*. [291] present the SPADE-E2VID neural network for event-based video reconstruction. They employ efficient sub-pixel convolutions [308] and spatially-adaptive denormalization (SPADE) [295] layers in upsampling blocks of the E2VID architecture. They use previously reconstructed images as the conditioning input of SPADE layers and transfer the style of these reconstructions to the current reconstruction. They show that this mechanism ensures temporal consistency without using an explicit loss function for it as in E2VID. They train this network with a synthetic dataset similar to E2VID's, using a *many-to-one* training scheme where the loss function is calculated only at the end of a sequence. They argue that the resulting network improves the contrast and quality of reconstructed videos, especially for the early frames during a test sequence.

Weng *et al*. [112] propose a hybrid CNN Transformer network for event-based video reconstruction, named Event Transformer Network (ET-Net). Specifically, they incorporated a Transformer [296] based module to the CNN-based encoder-decoder architecture of E2VID. The tokens for the transformer are extracted by the encoder layers in a multi-scale manner and then processed with the token pyramid aggregation module to model and output the global context of the event voxel grids. On the other hand, CNN-based components produce local information to complement this global context. The authors train this network with the same training set and data augmentations as in [289]. The resulting model improves the reconstruction quality at the expense of increased computational complexity.

Zhu *et al*. [292] propose a novel architecture named EVSNN, which is based on spiking neural networks. The architecture resembles E2VID in that it incorporates head, encoder, residual, decoder, and prediction layers. All layers except the prediction layer incorporate Leaky Integrate-and-Fire (LIF) neurons, while the prediction layer employs non-spiking Membrane Potential (MP) neurons to integrate all spikes and predict the grayscale image.

Furthermore, they propose an extension called potential-assisted EVSNN (PA-EVSNN), where they employ the Adaptive Membrane Potential (AMP) neuron, which adaptively updates the membrane time constant according to the input spikes. They consider EVSNN as a fully spiking network and the PA-EVSNN as a hybrid ANN-SNN network since the introduction of AMP neurons brings non-binary spikes in the network.

Instead of following the works that estimate intensity images from events via deep neural networks in an end-to-end manner, Zhang *et al*. [285] formulated the event-based image reconstruction task as a linear inverse problem based on optical flow. The optical flow is used to warp events and get the image of warped events (IWE). The IWE is then used in conjunction with the optical flow to solve a linear system of equations. The linear system is based on the event generation model that encode the relationships between derivatives of the brightness, events, and optical flow. This method avoids training deep neural networks and is more explainable compared to these end-to-end methods. Another benefit of this method is that it allows a natural extension to super-resolution. However, it has some negative aspects as well. First, the method requires the optical flow to be estimated first, and is adversely affected by errors in this estimation process. Second, it has hyperparameters which needs to be tuned to get the best performance in each scenario. Third, it can display artifacts due to camera motion, for certain cases like the center of rotation being on the image plane.

There are several concurrent works that tackle the task of event-based video reconstruction, each focusing on different aspects or methodologies. The work of Cho *et al*. [102] introduce a framework for joint learning of event-based object recognition and image reconstruction. They achieve this by using image-text embedding model CLIP [309] and specific loss functions to bridge the textual features of predicted categories and the visual features of reconstructed images. Zong *et al*. [310] propose a new event representation named Single Pixel Event Tensor (SPET). For each pixel, the number and polarity of events are counted for a given time interval to obtain 1-D tensors, which are then processed with a CNN based on 1-D convolutions. Although simple and fast, SPETs are affected via noisy events, and the global context of events is not leveraged for the reconstruction.

In [113], the authors propose a new framework where a second restoration phase is added after the first phase of image reconstruction, enhancing visual quality of reconstructed images. However, the restoration step is performed via Denoising Diffusion Probabilistic Models (DDPMs) [311], and computationally expensive. Liu and Dragotti [312] propose a model-based deep network, based on sparse coding of events and images, which is solved via iterative shrinkage thresholding algorithm (ISTA) [313]. The deep neural network is designed by unfolding iterations of ISTA via algorithm unfolding [314]. The work of Qu *et al*. [315] also propose a model-aided deep learning framework. They use a theory-inspired model which can generate video frames, and use a deep network to learn key parameters for the model.

3.2. Evaluation

Two main classes of evaluation done for intensity reconstruction methods are qualitative and quantitative image quality evaluation. Although some earlier works only present reconstructed images and rely on qualitative evaluation (*e.g*. [99]), most of the works also perform some form of quantitative evaluation. There are several metrics used in the literature to quantitatively evaluate reconstructed intensity images. These can be listed as MSE (mean squared error), PSNR (peak signal-to-noise ratio), BRISQUE score [4], FSIM (feature similarity index) [316], SSIM (structural similarity) [1] , and perceptual loss [2] (LPIPS). Some of these are full-reference metrics, while others are no-reference.

Full-reference metrics assess the quality of an image by comparing it to a reference (ground-truth) image that is assumed to be of perfect quality and measure the fidelity of the reconstructed image by quantifying visual differences between it and the reference. In contrast, no-reference metrics assess the quality of an image without any reference image, using models that predict perceived image quality based on typical image degradations and human visual perception.

Other than these image quality metrics, authors of [12] apply off-the-shelf frame-based computer vision algorithms on reconstructions and present a qualitative or quantitative

assessment for downstream tasks such as image classification, visual-inertial odometry, object detection, and monocular depth prediction.

3.2.1. Review of Evaluation Setups in Existing Work

There are several distinct evaluation methodologies in the literature, involving different datasets, event representations, post-processing steps, quantitative metrics, and downstream tasks. The absence of a standard evaluation procedure complicates fair comparison among performances of different methods. Furthermore, the specifics of these evaluation procedures are often not clearly outlined, even though every small detail can significantly alter the results, presenting challenges for reproducibility. Here, we review the evaluation setups used in existing event-based video reconstruction works, highlighting their diversity and identifying their shortcomings. This discussion provides the motivation for our proposed evaluation framework, EVREAL, detailed in Chapter 5.

The evaluation setup in [187] includes a small amount of data containing 1000 intensity frames taken from both real and simulated datasets, including the sequences from [99]. The authors compare their method against [99, 284] using sequences without ground truth frames and utilizing the no-reference metric BRISQUE [4]. The authors do not share their evaluation code.

Rebecq *et al*. [12, 87] use a selection of seven sequences from the ECD [280] dataset, using a fixed number of events to form event voxel grids and a tolerance of 1 ms to match the reconstructions with ground truth frames. To improve the output quality, they apply robust normalization as a post-processing step and then perform local histogram equalization before computing scores for MSE, SSIM, and LPIPS [2]. They compare their approach against [99] and [284]. They also report a temporal consistency score that requires a ground truth optical flow map between each frame. To obtain this, they use an off-the-shelf frame-based optical flow network [305], which has its own prediction errors. The researchers conduct experiments on challenging scenarios involving rapid motion, low-light conditions, and high dynamic range without providing any quantitative scores. Additionally, they report color

image reconstruction results from the event data available in the CED dataset [317] without providing any quantitative analysis.

Rebecq *et al*. also evaluate their method on four downstream tasks, including image classification, visual-inertial odometry, object detection, and monocular depth estimation [12, 87]. To perform these tasks, they feed reconstructed frames as inputs to task-specific frame-based methods and report either qualitative or quantitative results. For instance, for object classification, they use events from N-MNIST [318], N-Caltech101 [318], and N-Cars [152] datasets, and provide accuracy scores achieved by a ResNet-18 [15] network. Similarly, for visual-inertial odometry, they employ events from the ECD dataset and investigate mean translation errors obtained via VINS-Mono [319]. For object detection and monocular depth estimation, they use YOLOv3 [320] and MegaDepth [321], respectively, and only share qualitative results in a supplementary video. Additionally, they analyze the computational efficiency of their approach by reporting the frame synthesis time. The authors do not release their evaluation code publicly.

The evaluation setup of Scheerlinck *et al*. [91] mainly follows [12] and includes experiments on the selected frames from the sequences in the ECD dataset. They utilize a fixed number of events to form event voxel grids and apply local histogram equalization to reconstructions and ground truth frames before estimating quantitative metrics such as MSE, SSIM, and LPIPS. Additionally, they perform qualitative analysis on color image reconstruction and challenging scenarios involving high-dynamic range and fast motion. They focus on evaluating computational efficiency and compare several resolutions on GPU and CPU by examining the number of model parameters, memory consumption, FLOPs, and inference times. However, they do not conduct any downstream task experiments, and their evaluation codes are not publicly available.

Stoffregen *et al*. [289] evaluate their methods on a larger set of real-world sequences from three datasets, namely ECD [280], MVSEC [322] and their proposed HQF dataset. For ECD and MVSEC, they use the sequences commonly used in earlier work and report MSE, SSIM, and LPIPS scores. They always have a matching ground truth frame for each reconstruction

since they use events between each consecutive ground truth frame to form voxel grids (between-frames event grouping strategy). It is unclear whether they apply normalization or histogram equalization before calculating these scores. They do not perform experiments on challenging scenarios or downstream tasks, and they do not perform a computational efficiency analysis. The evaluation code is not publicly available.

Cadena *et al*. [291] evaluate their approach using seven sequences from the ECD dataset, starting from the very first frames of each sequence, and report MSE, SSIM, and LPIPS scores for quantitative comparison with E2VID and FireNet. They also introduce an RMS contrast metric to demonstrate that their method produces higher contrast reconstructions. To assess temporal consistency, they use a different off-the-shelf frame-based optical flow network [323] and report the corresponding scores. In addition, they perform object detection analysis on a single sequence of the ECD dataset, using events and YOLOv4 [324] to process reconstructed frames. They estimate ground truth object labels for two object classes by applying the same object detection network to ground truth intensity images and share average precision scores for this downstream task. They analyze the computational efficiency of their approach by reporting reconstruction time for inputs with various resolutions. While they released an evaluation code, we were not able to reproduce their results with it.

Weng *et al*. [112] conduct experiments using the ECD, MVSEC, and HQF datasets, with the same sequence cuts as in [289]. Events between consecutive ground truth frames are used to form voxel grids. To evaluate their approach, they calculate MSE, SSIM, and LPIPS scores without any normalization or histogram equalization applied to the reconstructed images. They compare their method with E2VID, E2VID+, FireNet, and FireNet+. Their supplementary material also includes qualitative results on challenging scenarios involving high-dynamic range and rapid motion. However, they do not perform a computational efficiency analysis or an experiment on a downstream task. The authors provide an open-source evaluation code, and we were able to reproduce their results with it.

The evaluation setup in [101] employs the ECD and HQF datasets, and the authors compare their method against E2VID, E2VID+, FireNet, and FireNet+. They also use between-frames

event grouping and employ local histogram equalization before calculating quantitative scores. They do not introduce a new architecture and thus do not perform a computational efficiency analysis. Qualitative results are also given for challenging scenarios such as high-dynamic-range and high-speed. No downstream task analysis is performed, and their evaluation code is not made publicly available.

In [292], the authors use ECD, MVSEC, and HQF datasets with between-frames event grouping and report quantitative scores using MSE, SSIM, and LPIPS metrics. They apply histogram equalization before calculating these scores. They compare their method with E2VID, E2VID+, FireNet, and SPADE-E2VID regarding image quality and computational efficiency. They also provide an analysis of energy consumption. However, they do not release an open-source evaluation code.

Zhang *et al*. [285] conduct a quantitative comparison with E2VID, E2VID+, and SSL-E2VID using MSE, SSIM, and LPIPS metrics. They focus on test sequences with limited camera motion, specifically selected from the ECD dataset, and also utilize events from N-Caltech101 [318] dataset. They align reconstructions with respective reference frames using Enhanced Correlation Coefficient Maximization [325]. They report median scores for each sequence instead of mean scores and present distribution plots of scores of each method on various sequences. They also analyze the effect of histogram equalization on quantitative scores and emphasize the importance of considering various factors while interpreting these scores. They showcase their method's ability to reconstruct color images and demonstrate temporal consistency on two example frames from the DSEC dataset [326]. They do not conduct experiments on downstream tasks or share their evaluation code.

3.3. Benchmark Datasets

There are several benchmark datasets proposed in the literature. Even when the methods are trained on large synthetic datasets (*e.g*. [12, 91, 289]), their performances are evaluated on datasets that are acquired with a real event camera. In this section, we present these

real-world datasets used in the literature for evaluating event-based video reconstruction methods. Table 3.2 presents a summary of these datasets.

Table 3.2 Benchmark Datasets

Ref	**Camera**	**Scenes**
CVPR16 [99]	DVS128	Indoor, dynamic, HDR.
ECD [280]	DAVIS240C	Indoor, outdoor. Various motions.
MVSEC [322]	DAVIS346B	Indoor, outdoor day/night. Various motions.
HS-HDR [12]	Samsung DVS Gen3	High speed, HDR. Many driving sequences.
CED [317]	Color-DAVIS346	Indoor, outdoor. Basic objects, people, driving.
HQF [289]	DAVIS240C	Indoor, outdoor. Various motions.

In our experimental work that we present in Chapter 7., we leverage additional datasets proposed for other tasks in the literature, such as event-based video frame interpolation [327] or visual-inertial state estimation [328]. In Section 7.1.1., we describe how we employ each dataset for our experiments. Furthermore, we propose a new dataset named HUE (the Hacettepe University Event dataset), present its details in Chapter 4., and use for our experiments in Chapter 7.

Now, we briefly review existing benchmark datasets currently being used for event-based image reconstruction methods in the literature:

Dataset of Bardow *et al*. [99] (CVPR16). This is one of the earliest datasets, captured indoors with a DVS128 camera, having a 128×128 pixel array. There are four sequences in the dataset, with one of them capturing a high-dynamic-range scene, and the others involve dynamic subjects like a person and a ball. [1]

Event Camera Dataset (ECD). This dataset is captured by a DAVIS240C sensor [71] where events and frames are generated from the same pixel array of 240×180 resolution. The common practice in the literature, established by Rebecq *et al*. [12], is to use seven short sequences from this dataset. These sequences mostly contain simple office environments with static objects, and the camera moves with 6-DOF and increasing speed.

[1] These sequences are available at `https://download.ifi.uzh.ch/rpg/web/data/E2VID/datasets/SOFIE_CVPR16/`

Multi Vehicle Stereo Event Camera (MVSEC) dataset. This dataset has longer sequences of indoor and outdoor environments captured by a pair of experimental mDAVIS-346B cameras. These cameras generate events and frames from the same pixel array with a 346×260 resolution. The indoor sequences are taken from a flying hexacopter, while the outdoor sequences are taken from a driving vehicle, during day and night times.

High-Speed and HDR datasets. These high-speed and HDR sequences are recorded by Rebecq *et al*. [12], using a Samsung DVS Gen3 event camera [329] with a spatial resolution of 640×480.

Color Event Camera Dataset (CED). This dataset consists of frames and events collected with a Color-DAVIS346 [111] camera, at 346×260 resolution. The camera has a color filter array (with RGGB Bayer pattern) and outputs color events and frames.

High-Quality Frames (HQF) dataset. The HQF dataset [289] contains fourteen sequences that exhibit various motion behaviors, including static, slow, and fast camera motion, and cover both indoor and outdoor scenes. Two different DAVIS240C cameras are used to capture the data, providing distinct noise and contrast threshold characteristics. The cameras generate events and intensity frames from the same 240×180 pixel array. The scenes and camera parameters are adjusted to ensure that the ground truth frames are well-exposed and have minimal motion blur.

4. PROPOSED DATASET

This chapter presents our proposed benchmark dataset HUE (the Hacettepe University Event dataset) by giving its motivation, the setup used to collect it, and the scenes and sequences it contains.

4.1. Introduction

While a growing number of event datasets are available, they come with certain limitations. Given these constraints, our motivation for presenting a new dataset is threefold. First, our dataset captures events at 1280×720, surpassing the resolution of all other datasets discussed in Section 3.3. Second, it features a substantial collection of sequences shot in diverse settings. Specifically, the HUE dataset includes 84 sequences recorded in both indoor and outdoor environments, using cameras that are either handheld or mounted on a vehicle and taken at various times of the day such as daylight, sunset, twilight, and nighttime. These sequences are recorded under various lighting conditions, including direct sunlight, shade, and artificial lighting, and they encompass a range of camera motions from slow to fast. The scenes also feature both static and dynamic objects, displaying people, animals, vehicles, buildings, everyday objects, cityscapes, and landscapes. Third, the HUE dataset specifically targets challenging low-light scenarios, with more than half of its sequences captured in conditions where the illuminance on the event sensor is just a few lux. In contrast, while the MVSEC dataset includes three nighttime sequences, they feature low-resolution events and are confined to driving scenarios.

Our dataset is collected with a setup consisting of two cameras: one event and one frame camera. Therefore our event sequences include complementary frames as well. As explained in Section 4.2.1., we employ a setup where the two cameras do not share the same optical axis and have different optical characteristics. Therefore, the event and frames that we collect are not pixel-wise spatially aligned. In our experimental analysis employing the HUE

dataset, we present these unaligned frames as a reference with qualitative results, and we use no-reference metrics for quantitative analysis.

Non-coaxial datasets like ours offer the advantage of simpler collection methods, as they do not require additional equipment such as optical beam splitters or specialized frame and event cameras. Due to these benefits, we anticipate that this type of camera setup will become increasingly common in the future [330].

In the remainder of this chapter, we present our data collection setup and the details of our collected dataset.

4.2. Data Collection Setup

As briefly mentioned above, one can choose from different types of optical setups to collect complementary event and frame data. One option is to use a camera that integrates both event-based and frame-based data collection mechanisms within the same pixel array, such as the Dynamic and Active Pixel Vision Sensor (DAVIS) [71]. This setup provides synchronized and pixel-wise aligned events and frames. However, this option comes with two significant disadvantages. Firstly, the shutter activity associated with each frame introduces noise into the event data, which is absent in sensors dedicated solely to event detection. Secondly, the most advanced model of this hybrid event-frame camera, the DAVIS346, only supports a maximum resolution of 346×260 pixels.

The second option involves using a setup that includes a beam splitter, an optical device featuring a 50/50 mirror that reflects half of the incoming light and transmits the other half. This arrangement allows frame and event sensors to share an aligned field of view. However, this setup has its disadvantages: it requires additional equipment, which incurs extra costs and occupies more space.

The third option, which we have chosen, is to use a stereo hybrid event-frame camera setup [330], where separate event and frame cameras are mounted side by side. This setup offers the advantage of simplicity as it does not require a beam splitter or a specialized

low-resolution camera like the DAVIS. In our data collection setup, the cameras are non-coaxial, each camera has a different resolution, and the lenses of each camera have differing fields of view. This configuration is the most generic setup possible and we anticipate it will become increasingly popular in the future due to its ease of integration into existing sensor suites, such as the multi-camera systems frequently seen in modern smartphones. The details of our data collection setup is presented in the following section.

4.2.1. Hardware and Optics

For our setup, we used a PROPHESEE Gen4M event camera [331] and an Allied Vision Alvium compact CMOS camera. The event camera features a Sony IMX636 event-based sensor. This sensor is in a $1/2.5''$ format with pixel dimensions of $4.86\mu m \times 4.86\mu m$, offering a resolution of 1280×720 and a dynamic range exceeding $120\,\text{dB}$. The event camera is equipped with a Soyo SFA 0820-5M lens, which has a fixed focal length of $8\,\text{mm}$ and a maximum aperture of $f/2.0$. The minimum focus distance is 0.1 meters, and the horizontal field of view is approximately 38 degrees.

The RGB camera contains a Sony IMX273 global shutter sensor. This sensor is in a $1/2.9''$ format with pixel dimensions of $3.45\mu m \times 3.45\mu m$, providing a resolution of 1456×1088, a dynamic range of $75\,\text{dB}$, and a 12-bit analog-to-digital converter. The RGB camera is paired with a Tamron M118FM06 lens, which has a fixed focal length of $6\,\text{mm}$ and a maximum aperture of $f/1.4$. The minimum focus distance is $0.1\,\text{meters}$, and the horizontal field of view is approximately 45 degrees.

The cameras are positioned with a baseline distance of approximately 2 cm between their optical axes, and their relative positions are kept fixed in an optical setup. For this optical arrangement, a custom mechanical part designed specifically for the cameras has been produced using a 3D printer. A picture of our setup is presented in Figure 4.1.

Considering that objects at different distances may be present in the scenes and need to be captured in focus, both lenses of the cameras have been set to a relatively narrow aperture

Figure 4.1 A picture showing our data collection setup.

of $f/8$, which provides a wide depth of field. Subsequently, the focusing distances of both lenses have been adjusted to the hyperfocal distance. This ensures that both cameras are set to capture all objects beyond a certain distance (for example, 0.5 meters) sharply.

For time synchronization of event and frame data, the RGB and event cameras have been directly connected to each other using a dedicated cable. This wiring involves connecting one of the general-purpose input/output (GPIO) lines of the RGB camera to one of the external trigger lines of the event camera. The software details of time synchronization are described in the following section.

4.2.2. Camera Settings and Software

The RGB camera has been configured to record at a frame rate of 25 frames per second with the specified exposure time. The PROPHESEE Gen4 event sensor, on the other hand, has six basic operating parameters named `bias_diff`, `bias_diff_on`, `bias_diff_off`, `bias_hpf`, `bias_fo`, and `bias_refr`. By adjusting these parameters, the sensor's sensitivity to positive and negative changes in light intensity, the cutoff frequencies of

low-pass and high-pass event filters, and the duration of the refractory period that each pixel must pass before generating another event after generating an event can be set. These settings ultimately affect the sensor's overall sensitivity, the temporal precision of generated events, the amount of background and noise events generated, and the time delay of events. To maintain a balance among these different characteristics in the sequences we collect, we have decided to leave the event sensor's parameters at their factory default settings.

Since we also record dynamic scenes with our setup, including moving objects, the exposure time of the RGB camera has been set to a maximum of $35\,\mathrm{ms}$ to minimize motion blur. The digital gain value of the RGB camera has been adjusted to a selected value based on the brightness level of the scene being recorded (higher for dark scenes, lower for bright scenes). However, increasing the digital gain value also results in images with higher noise levels. During data collection, we prefer images that are underexposed to a certain degree over images with higher noise. Therefore, the images we acquire with the RGB camera are generally underexposed in low-light scenarios.

We have developed software to complement our data collection hardware. This software runs on a computer and communicates with the RGB and event cameras via USB interfaces to set their configurations, control data acquisition of each camera, and receive acquired events and frames. After the mentioned settings are applied for both cameras, image frames from the RGB camera and event streams from the event camera are saved to the computer's permanent memory.

For time synchronization, the GPIO line of the RGB camera is programmed to be at a high voltage level (logic 1) during exposure and at a low voltage level (logic 0) during the remaining times. Similarly, the event camera continuously monitors the voltage level on its external trigger line and is programmed to record the rising and falling edges of the digital signal (transitions from logic 0 to 1 and from 1 to 0) separately and with high temporal precision. As a result, the moments when exposure begins and ends for each frame recorded by the RGB camera are timestamped with the high-resolution clock of the event camera. This allows the data from both cameras to be synchronized after recording.

The frames captured from the RGB camera are saved both as 12-bit raw images and as 8-bit 3-channel color images. The brightness change events captured by the event camera are recorded, including pixel position, polarity (increase or decrease in brightness), and a high-precision timestamp. The start and end times of each exposure of the frame camera are also recorded with high temporal precision by obtaining them from the event camera. Additionally, the amount of light falling on the event sensor, measured in lux, is recorded along with each frame.

4.3. Collected Dataset

As mentioned at the beginning of this chapter, our HUE dataset includes a large number of sequences captured in diverse and challenging scenarios. The dataset comprises 84 sequences, with an average duration of approximately 21 seconds, amounting to a total of nearly 30 minutes. It encompasses over 44,000 frames and approximately 40 billion events in total. Due to its large size and differing settings, we split the HUE dataset into six categories, loosely based on the scenes and various aspects. Here, we present these categories and the sequences in them:

HUE-City: This part of the dataset includes eight sequences, each recorded by looking outside from the window of the top floor of a mid-rise building, capturing cityscapes ahead. Five of these sequences are recorded during the daytime, while the remaining three are taken at twilight. The camera setup moves slowly, and the scenes are predominantly static, although there are occasional moving objects, such as vehicles or birds. Common elements like roads and buildings are mostly distant from the camera, with scenes extending towards the horizon. This enables us to test the limits of the spatial resolution of the event camera and evaluate the capability of each method in reconstructing fine details, as the scenes contain many objects and textures that are represented by just a few pixels on the sensor. Table 4.1 detail the sequences in HUE-City, by presenting the duration of each sequence in seconds, the number of frames and events, the illuminance on the event sensor measured in lux, and a short description. We also present sample frames and event visualizations in Figure 4.2. The

event visualizations are generated by accumulating events from a time interval on an image according to their polarities. The time interval for an event visualization, corresponding to a specific frame, is selected as the duration between the timestamps of that frame and the one preceding it. The timestamp of a frame is assumed to be in the middle of its exposure period. After accumulating events, we then apply normalization and a colormap to that image, such that events with positive and negative polarities are mapped to blue and red colors, respectively, and regions without events appear white.

Table 4.1 Breakdown of sequences in HUE-City. Columns 2 to 5 correspond to duration in seconds, number of frames, number of events in millions, and sensor illuminance level in lux.

Name	Dur. (s)	Fr.	Ev. (M)	Illum.	Description
city_day_1	14.4	361	124.8	33	Slow camera, birds fly
city_day_2	13.2	332	312.7	120	Slow camera, static scene
city_day_3	13.2	330	396.4	27	Close-by cars and trees
city_day_4	18.3	459	535.2	75	Close and far city
city_day_5	10.0	251	165.7	16	Several cars move on the road
city_twilight_1	21.9	549	372.9	1	Close-by moving car and far city
city_twilight_2	25.5	639	572.8	1	Apartments close and far, moving cars
city_twilight_3	22.3	557	447.1	1	Close-by cars and apartments, static

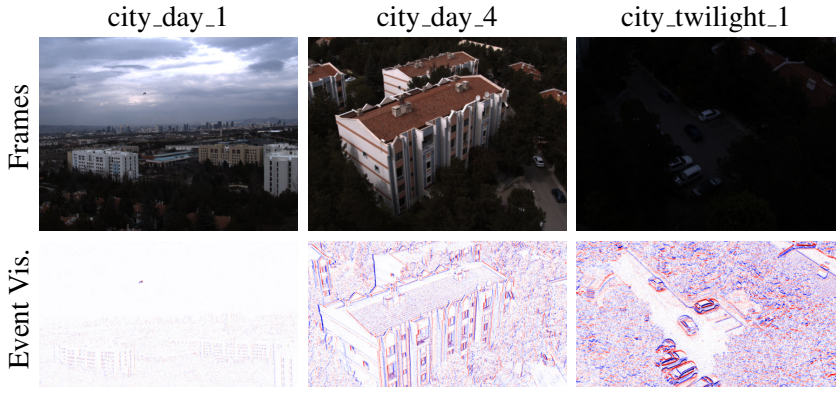


Figure 4.2 Sample scenes from HUE-City.

HUE-Day: This segment of the HUE dataset comprises 19 sequences, each captured during daylight, with a total duration of just over 7 minutes. While the majority of these sequences are filmed outdoors, there are two exceptions that are recorded in the atrium of a building bathed in natural sunlight. The scenes are predominantly dynamic, featuring

moving elements such as people, animals, vehicles, and tree leaves fluttering in the wind. The camera setup generally moves slowly, though it occasionally makes abrupt movements. The range of object distances varies widely, from close-ups of ants moving along the pavement to expansive views of buildings and trees stretching towards the horizon. This dataset serves the purpose of assessing reconstruction quality in well-lit and dynamic scenes. Details of these 19 sequences are given in Table 4.2, and sample scenes are presented in Figure 4.3.

Table 4.2 Breakdown of sequences in HUE-Day. Columns 2 to 5 correspond to duration in seconds, number of frames, number of events in millions, and sensor illuminance level in lux.

Name	Dur. (s)	Fr.	Ev. (M)	Illum.	Description
ants	19.8	495	618.8	24	Close-up to ants on the pavement
atrium_1	29.3	733	920.8	3	Atrium and courtyard, people walk
atrium_2	46.5	1164	1379.3	7	Coffeehouse, people stand & walk
cat	12.6	315	172.1	22	Cat watches around
construction	18.2	457	235.5	139	Sun and shade, construction
courtyard_1	47.9	1198	1125.5	11	People sit & walk
courtyard_2	26.9	674	790.0	13	Closer people
courtyard_3	18.4	460	397.3	6	People sit & walk
day_close_faces	17.6	442	442.3	34	Two close faces talk
day_dynamic_talk	12.3	309	304.1	46	Person talks dynamically
day_dynamic_walk	16.4	412	562.3	72	Person walks
day_shake_cat	15.1	378	555.1	27	Camera shakes and moves to cat
day_walk_cat	19.0	477	680.9	17	Camera and cat walk to each other
day_waving_leaves	19.4	485	199.3	96	Tree leaves wave, car moves
pidgeons_close	15.0	375	291.9	21	Pidgeons walking close-by
pidgeons_far_2	17.1	429	585.2	24	Pigeons walking far
sun_shade_building	23.8	595	523.0	9	Sun and shade on building
sun_shade_grass	10.4	260	192.6	27	Sun and shade, grass and building
terrace_pidgeon	41.2	1032	718.6	35	Terrace, cars, puddle, and pidgeon

HUE-Dark: The HUE-Dark is a subset of our dataset focused exclusively on low-light scenarios, where the average illuminance levels on the sensor are only a few lux. 18 of its sequences are taken outdoors during the twilight hours of the evening when the sun is below the horizon at varying degrees. More specifically, we define twilight hours as the period when the solar elevation angle (the angle of the sun's geometric center

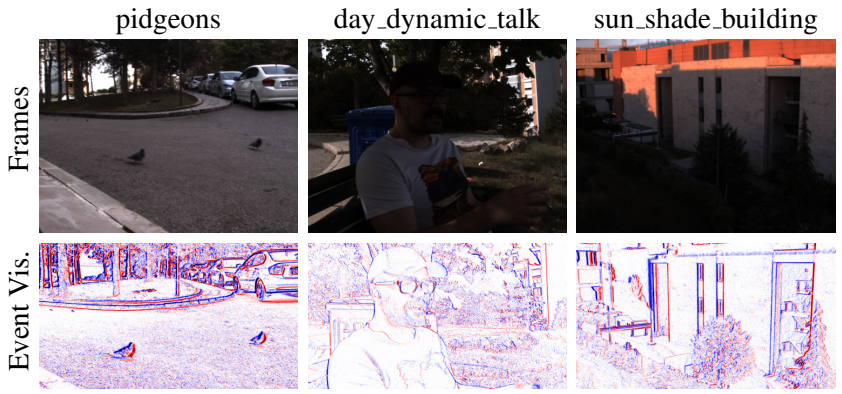


Figure 4.3 Sample scenes from HUE-Day.

relative to the horizon) is between 0 and -12 degrees, encompassing both civil twilight and nautical twilight. Additionally, three sequences are captured outdoors at night when the solar elevation angle is below -18 degrees, leading to significantly darker conditions. One final sequence is recorded in a very dark room, where a hand is waved in front of stationary cameras, illustrating minimal ambient lighting. In the outdoor twilight sequences, the primary source of illumination is sunlight scattering in the atmosphere, supplemented occasionally by artificial lights such as vehicle headlamps and streetlights. Conversely, in the outdoor night sequences, artificial lights become the primary source of illumination. This subset, comprising a total of 22 sequences, features a mix of natural and urban elements, ranging from lakes and forests to vehicles and buildings. Approximately one third of the sequences feature static scenes, while the remaining majority are dynamic, capturing movement within the environment. The details of HUE-Dark sequences and sample scenes are presented in Table 4.3 and Figure 4.4, respectively.

HUE-Indoor: This segment of our dataset, HUE-Indoor, includes 16 sequences captured in dimly lit indoor environments. Similar to the HUE-Dark subset, the sensor illuminance levels in these sequences are just a few lux, as can be seen from Table 4.4. This enables the evaluation of methods under low-light conditions but within indoor settings. Approximately half of these sequences are lit with natural light filtering in through windows, while artificial light sources illuminate the other half. One-third of the sequences feature moving objects, with the majority presenting static scenes. Some sequences are filmed in relatively

Table 4.3 Breakdown of sequences in HUE-Dark. Columns 2 to 5 correspond to duration in seconds, number of frames, number of events in millions, and sensor illuminance level in lux.

Name	Dur. (s)	Fr.	Ev. (M)	Illum.	Description
car	2.9	73	9.2	0	Car passes ahead
dark_equipment	20.3	507	163.9	0	Construction equipment in the dark
dark_forest_1	30.4	760	551.1	0	Dark forest
dark_forest_2	11.3	282	177.7	0	Dark forest and illuminated path
duck_fence	15.1	379	266.6	0	Ducks behind fence, text
duck_fence_lake	22.7	569	811.9	0	Lake and ducks behind fence
duck_lake_1	26.1	654	647.4	2	Duck standing close besides lake
duck_lake_2	38.9	974	1322.5	1	Ducks moving, waves on lake
duck_lake_3	28.8	721	849.7	0	Wavy lake and ducks
duck_lake_4	35.6	891	552.7	0	Ducks swim in the dark lake
lake_1	18.3	459	468.8	3	Lake, mostly static
lake_2	13.6	340	310.0	1	Closed cafe, lake with reflections
lake_3	14.8	370	275.6	1	Lake with reflections, static scene
lake_4	23.8	595	426.1	1	Lake with reflections, mostly static
night_parking_lot	23.9	598	148.1	0	Person walks at night
person_face	7.8	196	63.6	0	Person face under lamp at night
person_walk	10.2	255	25.3	0	Person walks under lamp at night
road_field	25.2	630	1374.7	2	Car passes, men playing football
sunset_parking_lot	24.6	615	408.4	3	Parking lot, sun is down, cars pass
terrace_sunset	21.9	548	715.2	5	Terrace, laptop, reflection, sunset
very_dark_hand	3.8	96	22.8	0	Very dark, static camera, hand wave
water_flow	26.5	662	489.0	0	Water flow, camera slowly moves

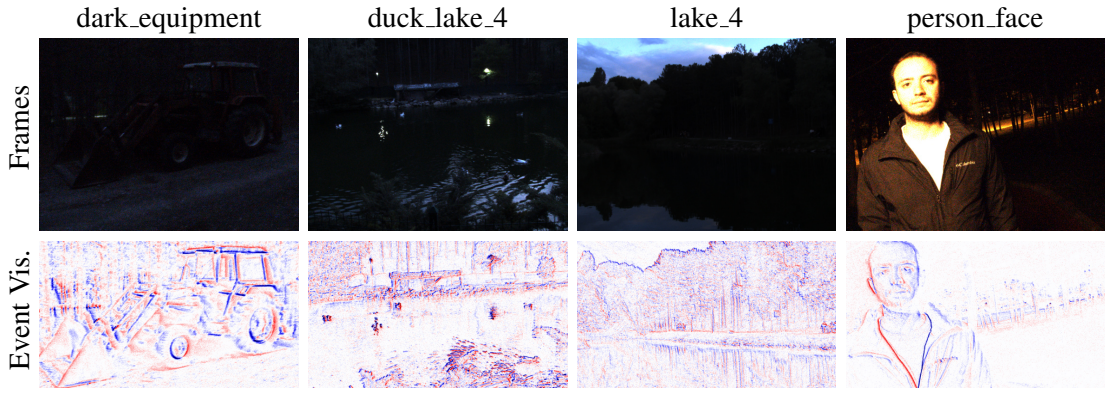


Figure 4.4 Sample scenes from HUE-Dark.

large indoor halls and corridors, featuring distant objects, whereas most focus on closer subjects. Sample scenes presenting these aspects are displayed in Figure 4.5 via frames and event visualizations. HUE-Indoor is particularly valuable for assessing the reconstruction performance on objects with fine details, such as small texts and textured regions, under low-light conditions.

Table 4.4 Breakdown of sequences in HUE-Indoor. Columns 2 to 5 correspond to duration in seconds, number of frames, number of events in millions, and sensor illuminance level in lux.

Name	Dur. (s)	Fr.	Ev. (M)	Illum.	Description
bookshelves	14.4	361	542.9	0	Indoor bookshelves
atlas	9.6	241	345.4	0	Large public library hall, people move
corridor	26.1	654	593.0	3	Walk in corridor, dynamic person
dome	7.4	184	115.7	0	Dome of the hall and writings on it
figures_classics	18.7	467	628.8	0	Shelves close up, figures and CDs
lab_1	4.9	123	194.1	0	Dim laboratory, short sequence
lab_2	34.7	869	1340.7	1	Dim laboratory, longer sequence
laptop	8.4	210	244.0	1	Laptop screen, text in command line
letters	9.4	234	355.1	0	Display of hanging letters, large book
miniature	8.8	220	156.2	0	Display of old miniature paintings
old_books	8.4	211	134.3	0	Display of old books
old_classroom	17.1	427	259.3	0	Display of old classroom, static scene
posters_window	32.1	803	1050.2	0	Posters indoors, people walk outdoors
recycle_art	7.2	179	133.8	0	Display of recycle art, writing
selfie	7.9	197	69.6	1	Short indoor face sequence
stairways	11.5	287	215.6	0	Stairways, people walk

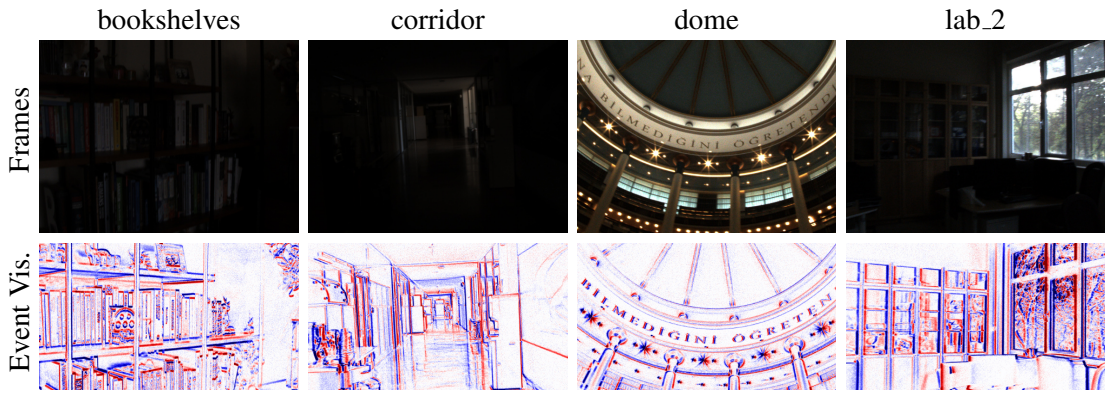


Figure 4.5 Sample scenes from HUE-Indoor.

HUE-Drive: This portion of the HUE dataset comprises 12 driving sequences, where the camera setup is mounted inside of a vehicle's front windshield, monitoring ahead through this window. Throughout these recordings, the vehicle travels through various street settings in daylight, twilight, and nighttime, capturing elements such as other cars, pedestrians, motorbikes, parked vehicles, gas stations, tunnels, interchanges, and roundabouts. The descriptions of each sequence, as well as other details, are presented in Table 4.5. Two of the sequences are captured in daylight, while the remaining ten are equally divided between twilight and nighttime, with five in each lighting condition. This subset is important for evaluating performance in dynamic scenes and under challenging lighting conditions, such as dark roads and rapidly moving headlights. Example scenes from HUE-Drive are presented in Figure 4.6.

Table 4.5 Breakdown of sequences in HUE-Drive. Columns 2 to 5 correspond to duration in seconds, number of frames, number of events in millions, and sensor illuminance level in lux.

Name	Dur. (s)	Fr.	Ev. (M)	Illum.	Description
drive_day_1	25.5	639	763.1	43	Sunny street, cars and people move
drive_day_2	27.3	683	740.4	43	Sunny and shady street, a van moves
drive_night_1	39.0	975	285.7	0	Through a tunnel with other cars
drive_night_2	38.3	958	354.4	0	Through interchange and roundabout
drive_night_3	42.2	1055	285.0	0	Turning in dark streets
drive_night_4	6.0	151	68.0	0	Cars move in opposite direction
drive_night_5	21.1	527	97.9	0	Very dark road, no other cars
drive_twilight_1	23.9	598	231.3	0	Driving slow, a motorcycle passes
drive_twilight_2	24.6	615	339.7	0	Driving in the street behind a car
drive_twilight_3	34.9	872	358.0	0	Passing by parked vehicles
drive_twilight_4	27.0	675	181.5	0	Driving on the road, several cars
drive_twilight_5	32.7	817	296.2	0	Through interchange and roundabout

HUE-HDR: This final subset of the HUE dataset focuses exclusively on high-dynamic range scenarios, where significant changes in illumination levels occur within the sequences. Details of the seven sequences in this subset are given in Table 4.6. The ratio between the illuminance levels of highly lit and low-lit parts of the scene often varies by orders of tens or hundreds. This aspect tests the high-dynamic range properties of event streams and images

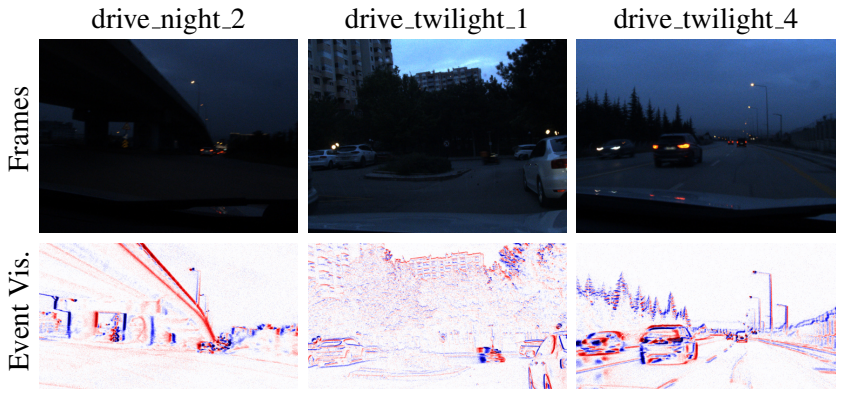


Figure 4.6 Sample scenes from HUE-Drive.

reconstructed with them, when compared to frames acquired by the low-dynamic range frame camera, as can be seen in frames displayed in Figure 4.7.

Table 4.6 Breakdown of sequences in HUE-HDR. Columns 2 to 5 correspond to duration in seconds, number of frames, number of events in millions, and sensor illuminance level in lux.

Name	Dur. (s)	Fr.	Ev. (M)	Illum.	Description
hdr_atrium	38.9	973	1168.4	8	Coffeehouse, people walk and sit
hdr_courtyard	28.8	720	876.8	24	Camera rotates, people walk and sit
hdr_plants	20.6	516	601.9	14	Indoor plants and window city view
hdr_selfie	10.2	255	173.6	18	Indoor face, window with daylight
hdr_sun	17.5	437	568.6	785	Abrupt motion, viewing cat and sun
hdr_terrace_sun_1	42.4	1060	969.9	315	Views of trees, vehicles, and sun
hdr_terrace_sun_2	32.3	807	1032.8	24	Terrace walk, reflections, and sun

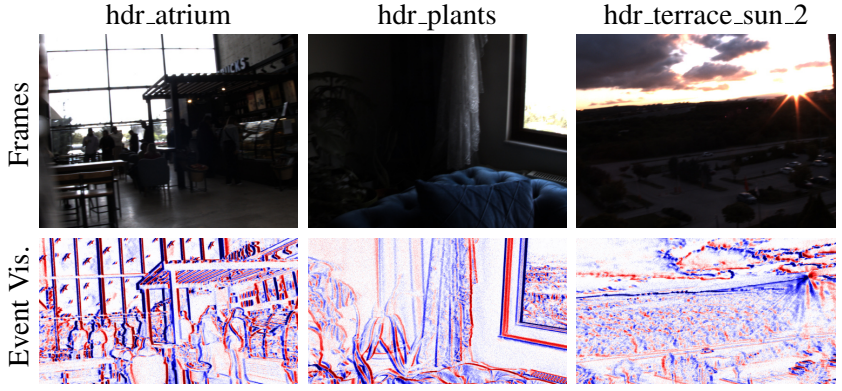


Figure 4.7 Sample scenes from HUE-HDR.

5. EVREAL: TOWARDS A COMPREHENSIVE BENCHMARK AND ANALYSIS SUITE FOR EVENT-BASED VIDEO RECONSTRUCTION

This chapter presents EVREAL, our proposed open-source evaluation framework for event-based video reconstruction, based on our published work [332],

- Burak Ercan, Onur Eker, Aykut Erdem, and Erkut Erdem. EVREAL: Towards a comprehensive benchmark and analysis suite for event-based video reconstruction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pages 3942–3951. 2023.

In this work, I have contributed to the conceptualization, design of evaluation methodology, implementation of the framework, and experimental analysis of image quality, robustness, and computational complexity; while Onur Eker contributed more to the implementation and evaluation of downstream tasks and visualization.

In this chapter, we present the details of our EVREAL framework, including its features and various experimental analyses that can be performed with it, while the experimental details and results are presented in Chapter 7. The code for EVREAL is available at `https://github.com/ercanburak/EVREAL`.

5.1. Introduction

While recent deep learning-based methods have shown promise in video reconstruction from events, this problem is not completely solved. Comparing different approaches via evaluation protocols is essential to procure progress in this task. Thus, a significant effort has been put forth to find better ways to evaluate event-based video reconstruction methods and assess the visual qualities of reconstructed videos. There are several distinct evaluation

methodologies involving different datasets, event representations, post-processing steps, quantitative metrics, and downstream tasks (Section 5.1.2. presents an overview, and a more detailed discussion can be found in Section 3.2.1.). However, the lack of a standard evaluation procedure makes it hard to compare the performances of different methods fairly. The details of the evaluation procedures are sometimes not clearly defined, even though each minor detail may significantly alter the results. This also poses challenges for other researchers to reproduce the reported results. This motivates the need for open-source codes and standardized protocols for evaluation.

5.1.1. Challenges of Evaluation

Evaluation, although an essential part of works on video reconstruction from events, has challenges on its own. In this section, we outline these challenges, which serve as the motivation for our evaluation framework.

Comparing different methods requires not only well-defined evaluation protocols but also a diverse set of test datasets that cover various real-world settings. Large-scale benchmarks have been instrumental in advancing many computer vision tasks, as demonstrated by ImageNet [115] for image classification and MS-COCO [114] for object detection, providing results that are generalizable to unseen real-world data. However, since event-based vision is a relatively new field compared to classical frame-based computer vision, the current datasets used for assessing event-based video reconstruction are limited in scale and scope, confined to specific domains, scenes, camera types, and motion patterns. To ensure the generalizability of the results and evaluate the methods' effectiveness in more real-world scenarios, it is essential to assess their performances on a large variety of datasets showing different characteristics.

Event data is handy in scenarios where traditional frame cameras fail, such as scenes captured under low-light conditions or with rapid motion and underexposed or overexposed regions. Hence, it is of utmost importance to evaluate the effectiveness of event-based video reconstruction models in those challenging situations. However, as traditional frame-based

cameras especially struggle in these scenarios, collecting high-quality reference frames is a challenging task on its own. This paradox makes it difficult to quantify the success of event-based video reconstruction methods where they are most needed.

Even in scenarios where it is possible to collect high-quality reference frames with minimal motion blur and optimal exposure, assessing image quality remains a subjective endeavor. Hence, the current studies generally consider a perceptual metric like learned perceptual image patch similarity (LPIPS) [2] along with distortion-aware metrics like PSNR and structural similarity (SSIM) [1]. However, as a full-reference metric, LPIPS is trained on distortions that are not commonly seen in the reconstructed intensity images from event data. Hence, this raises some doubts about the significance of these perceptual comparisons.

Reconstructing images from events is a complex task. It depends on many variables that can affect the performance of the methods. These include sensor noise characteristics, sensor parameters, event generation rate, event grouping scheme, grouping rate, frame reconstruction rate, and temporal regularity. Despite their importance, the literature often overlooks the robustness of the methods to changes in these variables. Therefore, it is crucial to evaluate the sensitivity of the methods to these variables and to consider their performance under changing conditions. A method that performs well under specific settings may not be suitable for general use when these variables are expected to change.

Event cameras are known for their low-latency and non-redundant data flow, making them ideal for scenarios that require real-time and low-power processing. As a result, the computational efficiency of event-based video reconstruction methods is just as important as the visual quality of reconstructions. Neglecting this aspect in a benchmark could lead to choosing a method that provides high-quality reconstructions but is impractical for real-time processing.

Table 5.1 A comparison of our proposed EVREAL framework to the experimental evaluation setups reported in the existing work in terms of datasets being used, methods compared, number of reconstructed frames used in quantitative analysis, and metrics being utilized. We also indicate whether each evaluation setup includes analysis of computational efficiency, challenging scenarios (fast motion, low light, or high-dynamic range), downstream tasks, and robustness. Finally, we mark whether the implementation of this evaluation setup is open-sourced or not. In the metrics column, FR and NR stand for full-reference and no-reference metrics, respectively. M:MSE, S:SSIM [1], L:LPIPS [2], and T:Temporal Consistency [3] are the full-reference metrics, while R:RMS contrast, B:BRISQUE [4], N:NIQE [5], and M:MANIQA [6] are the no-reference metrics. In the Challenging Scenarios, the Downstream Tasks and Robustness Experiments columns, each 🖼 symbol denotes a reported qualitative analysis and a 📊 symbol represents a quantitative analysis being performed along with a qualitative comparison.

Evaluation Setup in	Test Datasets	# of Frames	Compared Methods	Metrics (FR/NR)	Comp. Eff.	Chlng. Scnrs.	Downstream Tasks	Robustness Expermnts.	Open Source
[187]	[99]	0.2K	[99, 187, 284]	–/B		🖼			
[12]	[280]	1.9K	[12, 99, 284]	MSLT/–	✓	🖼	🖼 🖼 📊 📊	🖼	
[91]	[280]	1.9K	[12, 91, 99, 284]	MSL/–	✓	🖼			
[289]	[280, 289, 322]	28.7K	[12, 91, 289]	MSL/–					
[291]	[280]	3.1K	[12, 91, 291]	MSLT/R	✓		📊		✓
[112]	[280, 289, 322]	28.7K	[12, 91, 112, 289]	MSL/–		🖼			✓
[101]	[280, 289]	17.4K	[12, 91, 101, 289]	MSL/–		🖼			
[292]	[280, 289, 322]	28.7K	[12, 91, 289, 291, 292]	MSL/–	✓			📊	
[285]	[280, 318]	1.9K	[12, 101, 285, 289]	MSL/–	✓	🖼		📊	
Ours	[12, 280, 289, 322, 327]	47.7K	[12, 91, 101, 112, 289, 291]	MSL/BNM	✓	📊	📊 📊 📊	📊	✓

5.1.2. EVREAL

To address these issues and facilitate progress in event-based video reconstruction, in this chapter, we propose EVREAL, Event-based Video Reconstruction Evaluation and Analysis Library, an open-source framework based on PyTorch [333]. Our framework offers a unified evaluation pipeline to benchmark pre-trained neural networks and a result analysis tool[2] to visualize and compare reconstructions and their scores.

Using EVREAL, it is possible to use a large set of real-world test sequences and various full-, and no-reference image quality metrics to perform qualitative and quantitative analysis under diverse conditions, including challenging scenarios such as rapid motion, low light, and high dynamic range. In Chapter 7., we present the details of experiments we conduct with EVREAL, their results, and insightful observations. Moreover, we conduct experiments to assess the performance of each method under variable conditions and analyze their robustness to these varying settings. We also evaluate the quality of video reconstructions via downstream tasks like camera calibration, image classification, and object detection. This extrinsic evaluation can be considered a proxy metric for image quality or a task-specific metric if the goal of event-based video reconstruction is to perform these downstream tasks.

In Section 5.1.2., we present an overview of our experimental setup in comparison to prior work. Along with EVREAL, we build a website to share our results and findings, together with the source code to reproduce them [3]. We also intend to update this webpage on a regular basis as new event-based video reconstruction methods are proposed. Our contributions in this chapter can be summarized as follows:

- We propose a unified evaluation methodology and an open-source framework to benchmark and analyze event-based video reconstruction methods from the literature.
- Our benchmark includes additional datasets, metrics, and analysis settings that have not been reported before. In Chapter 7., we present quantitative results on challenging

[2] https://ercanburak-evreal.hf.space
[3] https://ercanburak.github.io/evreal.html

scenarios involving rapid motion, low light, and high dynamic range. Moreover, we conduct additional experiments to analyze the robustness of methods under varying settings such as event rate, event tensor sparsity, reconstruction rate, and temporal irregularity.

- To further examine the quality of the reconstructions, EVREAL provides quantitative analysis on several downstream tasks, including camera calibration, image classification, and object detection.

5.2. Task Description

Let us assume that we have an event stream $\{e_i\}$ consisting of N_E events that span a duration of T seconds. Each event $e_i = (x_i, y_i, t_i, p_i)$ in the stream represents a change in brightness perceived by the sensor, and contains information about the pixel location (x_i, y_i), the timestamp t_i, and the polarity p_i of this intensity change. Here, $t_i \in [0, T]$, $p_i \in \{+1, -1\}$, $x_i \in \{0, \dots, W-1\}$, and $y_i \in \{0, \dots, H-1\}$ for all $i \in \{0, \dots, N_E - 1\}$, with W and H denoting the width and height of the sensor array, respectively.

Given only these events, our task aims to generate an image stream $\{\hat{I}_k\}$ of N_I images from that same time period of T seconds. Each image $\hat{I}_k \in [0, 1]^{W \times H}$ is a 2D grayscale representation of the absolute brightness of the scene as if captured by a standard frame-based camera at some time $s_k \in [0, T]$ for all $k \in \{1, \dots, N_I\}$.

It is important to note that we constrain our task description such that each generated image only depends on past events, *i.e.* only $\{e_i \mid t_i \leq s_k\}$ is used to generate an image $\hat{I}_k$. This allows our method to be used in scenarios where future events are not observed yet, such as reconstructing intensity images from a continuous event camera stream in real-time.

5.3. Proposed Evaluation Framework and Pipeline

EVREAL implements several standardized components crucial for deep event-based video reconstruction models, including *event pre-processing*, *event grouping*, *event representation*,

representation processing, and *image post-processing* (see Figure 5.1). We have included components to evaluate the visual quality of each frame in the generated videos, which are split into *full-reference* metrics and *no-reference* metrics. The former is utilized when high-quality, distortion-free ground truth frames are available. In contrast, the latter is used when ground truth frames are of low quality or unavailable (refer to Section 5.5.). To assess the practical use of a given method, our framework allows for evaluating it on several downstream tasks. Specifically, we analyze the performance of tested models on three downstream tasks, *object detection*, *image classification*, and *camera calibration* (Section 5.8.).

EVREAL also includes an analysis tool. Given a set of reconstructions generated by one or more methods, it collects ground truth frames, event visualizations, event rate statistics, and instantaneous values for a set of quantitative metrics. It then generates an output video that displays this data in a time-synchronized manner, including plots of quantitative metrics. The tool can be used online at `https://ercanburak-evreal.hf.space`. Our tool is particularly valuable in pinpointing specific limitations and failure cases of methods. For instance, it can reveal situations where noisy reconstructions significantly impact future reconstructions due to the sequential nature of the method. Such scenarios can be visually identified from the plots of quantitative metrics.

In the following, we provide detailed descriptions of the components of our evaluation framework.

Event pre-processing. This component can be employed to process raw events before grouping them. Possible pre-processing operations include event temporal downsampling and adding artificial event noise to perform robustness experiments under these conditions.

Event grouping. Each event in isolation contains limited information about the scene, so a common practice is to group events together and process them as a whole. We consider event groups i) with a fixed number of events, ii) spanning fixed duration, and

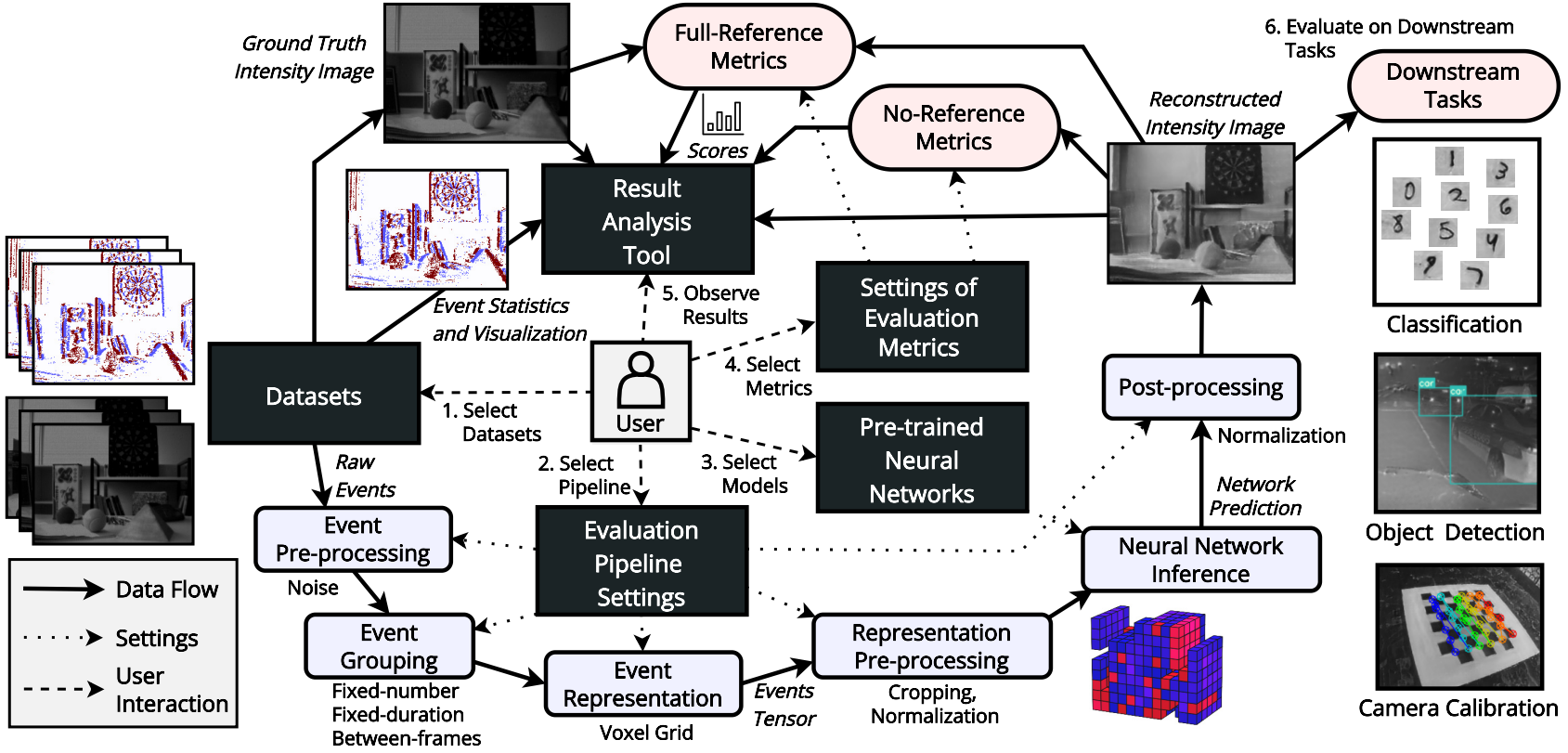


Figure 5.1 An overall look at our proposed EVREAL (Event-based Video Reconstruction – Evaluation and Analysis Library) toolkit.

iii) between consecutive frames. The details of these event grouping strategies are presented in Section 2.2.2.

Event representation. There are several event representations in event-based vision literature, which we briefly overview in Section 2.2.1. A common choice for utilizing deep CNN architectures for event-based data is to accumulate grouped events into a grid-structured representation such as a voxel grid. Specifically, the voxel grid event representation presented in [201] is used in every event-based video reconstruction method that we incorporate in EVREAL, including our proposed method, HyperE2VID. Thus, we implement this event representation in EVREAL, and present its details in Section 6.3.1.

Representation pre-processing. After forming a representation from grouped events, it is possible to pre-process this representation before feeding it to the neural network, such as cropping or applying normalization. During our experimental analysis, we apply such pre-processing steps when required by the respective method, as described in Chapter 7.

Neural network inference. This module is used for predicting intensity frames given the event representation by employing the pre-trained neural network model chosen by the user. As mentioned earlier, we use PyTorch here.

Post-processing. It is also possible to post-process the intensity frame that the network predicts, by utilizing procedures like robust min/max normalization, as done in [12]. While performing our experiments, we also apply any post-processing operations as suggested by each method. The details of those are given in Chapter 7.

5.4. Compared Methods

In EVREAL, we include eight methods from the literature that have PyTorch-based open-source model codes and pre-trained models. These methods include E2VID [12], FireNet [91], FireNet+ and E2VID+ [289], SPADE-E2VID [291], SSL-E2VID [101],

and ET-Net [112], as well as our proposed HyperE2VID [334]. Note that E2VID+ and SSL-E2VID share the same deep network architecture as E2VID, while FireNet+ employs the same architecture as FireNet. Here, we utilize the pre-trained models shared publicly by the authors and evaluate them on the same datasets under a common experimental evaluation setup.

5.5. Quantitative Image Quality Metrics

To quantitatively assess the quality of videos reconstructed from events, we use both full-reference and no-reference metrics. Full-reference metrics, as their name implies, provide a quality score for an image in regard to a given reference image. In contrast, no-reference metrics do not require any ground truth image and give perceptual quality scores by directly processing input images.

In our experimental procedure depicted in Chapter 7., we utilize three full-reference evaluation metrics: MSE, SSIM [1], and LPIPS [2]. These metrics are employed only when high-quality, distortion-free ground truth frames are available. While the MSE and SSIM are better suited for capturing distortions, LPIPS measures the perceptual similarity by a deep network trained to conform with human visual perception. Furthermore, we utilize three no-reference metrics: BRISQUE [4], NIQE [5], and MANIQA [6]. These metrics are used when the ground truth frames are of low quality or when they are not available at all. BRISQUE and NIQE are traditional metrics that employ hand-crafted features and measure conformity to natural scene statistics, considering various synthetic and authentic distortions such as blur, noise, and compression. On the other hand, MANIQA is a deep-learning based method employing vision transformer architecture [47], which is trained in an end-to-end manner to assess perceptual image quality while specifically focusing on distortions seen in the outputs of neural network based image restoration algorithms. The results of the aforementioned metrics can be influenced by specific settings. Hence, to ensure consistency, we provide the detailed settings that we use in Section 7.1.2.

Furthermore, we would like to mention that our open-source EVREAL framework fully supports the IQA-PyTorch toolbox [335], so that all metrics in that toolbox can be directly used in EVREAL to assess quality of reconstructions[4].

5.6. Datasets

EVREAL supports event datasets as long as they are converted to the required format. In our open-source repository, we share scripts to perform conversion from common formats such as rosbag and text files. Furthermore, we share scripts and instructions to download and convert datasets such as ECD [280], MVSEC [322], HQF [289], and N-Caltech101 [318]. A list of all the datasets that we employ in our experiments and a discussion on how we use each dataset can be found in Section 7.1.1.

5.7. Color Reconstruction

EVREAL also supports color reconstructions using the CED dataset [317], which is collected with the Color-DAVIS346 camera. To generate color reconstructions, we adopt the method described in [12]. This involves reconstructing each color channel separately at quarter resolution, then upsampling and merging them to form a full-color image. Next, we convert this image to LAB color space and replace its luminance channel with a high-resolution grayscale reconstruction derived from all events. Results can be seen in Section 7.3.

5.8. Analysis on Downstream Tasks

Event cameras, due to their unique characteristics, can provide a viable alternative to traditional frame-based cameras in challenging conditions. As a result, using images reconstructed from event streams for downstream tasks when standard cameras fail can be beneficial. To assess the effectiveness of each method in an extrinsic manner, we leverage downstream computer vision tasks including object detection, image classification, and

[4]There are a total of 64 such metrics as of v0.1.10 of IQA-PyTorch toolbox.

camera calibration. We give a detailed description of our experimentation with these tasks below, while the results of them are presented in Section 7.6.

Object detection. Object detection is a significant area of research in computer vision, with numerous applications ranging from autonomous navigation to medical imaging. However, traditional frame-based cameras often fail to capture satisfactory images under low-light conditions, affecting the performance of object detection methods. Since event cameras have a high dynamic range compared to traditional cameras, we evaluated the performance of object detection on low-light images captured by event cameras. For this purpose, we used the MVSEC-NIGHTL21 car detection dataset [336], derived from the $outdoor\_night1\_data$ sequence of MVSEC, captured under night driving conditions. The dataset contains 2,000 labeled intensity images, with 1,600 frames for training and 400 frames for validation. We reconstructed images from the provided event sequence for each method and extracted frames corresponding to those in the MVSEC-NIGHTL21 dataset. We then used YOLOv7 [337] object detector to detect cars in the reconstructed images and intensity images of the frame camera. We used a model trained on the MS-COCO dataset [114] for car detection in the images and evaluated the results using the PASCAL VOC metric [338], providing the AP score for each method on the dataset.

Image classification. We evaluated the performance of our image reconstruction methods using two image classification datasets: Neuromorphic-Caltech101 (N-Caltech101)[318] and Caltech101[339]. N-Caltech101 is a spiking version of the original Caltech101 dataset, containing 100 object classes plus a background class (excluding the "Faces" class). We trained a ResNet50 [15] classification model on Caltech101, excluding the "Faces" class to ensure consistency between the datasets. For each method, we reconstructed images on event streams from N-Caltech101, and ran the ResNet50 model on the reconstructions to evaluate their accuracy on the dataset.

Camera calibration. It is a critical component of computer vision systems, but traditional calibration techniques for standard frame-based cameras cannot be applied to event cameras due to their asynchronous pixel output. Recently, Muglikar *et al*. [95] demonstrated that

image reconstruction can be used to apply conventional calibration techniques for accurate event-camera calibration. In this study, we compare the performance of various image reconstruction methods for camera calibration using the *calibration* sequence from the ECD dataset. This sequence consists of an event camera moving in front of a calibration target, and the intrinsic calibration parameters of the DAVIS240C, provided by ECD, serve as the ground truth. We reconstruct image sequences using each method and accordingly obtain intrinsic calibration parameters using the reconstructed images and the `kalibr` toolbox [340]. We then measure the mean absolute percentage error (MAPE) of the intrinsic calibration parameters to determine the most effective method.

6. HyperE2VID: IMPROVING EVENT-BASED VIDEO RECONSTRUCTION VIA HYPERNETWORKS

In this chapter, we present our event-based video reconstruction method HyperE2VID, which employs a novel dynamic neural network architecture via hypernetworks and improves the current state-of-the-art methods in terms of both image quality and efficiency. This chapter is based on our published work [334],

- Burak Ercan, Onur Eker, Canberk Saglam, Aykut Erdem, and Erkut Erdem. HyperE2VID: Improving event-based video reconstruction via hypernetworks. IEEE Transactions on Image Processing, 33:1826–1837, 2024.

In this work, I have contributed to the network design and implementation, data curation and generation, experimentation, and analysis. Onur Eker partly contributed to the implementation of the network and experimental analysis, while Canberk Saglam partly contributed to some of the ablation studies and qualitative analysis.

We present specifics related to our methodology in this chapter, while a detailed experimental study is presented in Chapter 7. The code for HyperE2VID is available at `https://github.com/ercanburak/HyperE2VID`.

6.1. Introduction

Recently, deep learning based methods have obtained impressive results in the task of video reconstruction from events (*e.g.* [87, 112, 289]). To use successful deep architectures in conjunction with event-based data, these methods typically group events in time windows and accumulate them into grid-structured representations like 3D voxel grids through which the continuous stream of events is transformed into a series of voxel grid representations. These grid-based representations can then be processed with recurrent neural networks (RNNs), where each of these voxel grids is consumed at each time step.

Since events are generated asynchronously only when the intensity of a pixel changes, the resulting event voxel grid is a sparse tensor, incorporating information only from the changing parts of the scene. The sparsity of these voxel grids is also highly varying. This makes it hard for neural networks to adapt to new data and leads to unsatisfactory video reconstructions that contain blur, low contrast, or smearing artifacts ([87, 289, 291]). Recently, Weng *et al*. [112] proposed to incorporate a Transformer [296] based module to an event-based video reconstruction network in order to better exploit the global context of event tensors. This complex architecture improves the quality of reconstructions, but at the expense of higher inference times and larger memory consumption.

The methods mentioned above try to process the highly varying event data with *static* networks, in which the network parameters are kept fixed after training. Concurrently, there has been a line of research that investigates *dynamic* network architectures that allow the network to adapt its parameters dynamically according to the input supplied at inference time. A well-known example of this approach is the notion of *hypernetworks* [341], which are smaller networks that are used to dynamically generate weights of a larger network at inference time, conditioned on the input. This dynamic structure allows the neural networks to increase their representation power with only a minor increase in computational cost [342].

In this chapter, we present our proposed method HyperE2VID, which improves the current state-of-the-art in terms of image quality and efficiency (see Figure 6.1) by employing a dynamic neural network architecture via hypernetworks. Our proposed model utilizes a main network with a convolutional recurrent encoder-decoder architecture, similar to E2VID [87]. We enhance this network by employing dynamic convolutions, whose parameters are generated dynamically at inference time. These dynamically generated parameters are also spatially varying such that there exists a separate convolutional kernel for each pixel, allowing them to adapt to different spatial locations as well as each input. This spatial adaptation enables the network to learn and use different filters for static and dynamic parts of the scene where events are generated at low and high rates, respectively. We design our hypernetwork architecture in order to avoid the high computational cost of generating per-pixel adaptive filters via filter decomposition as in [343].

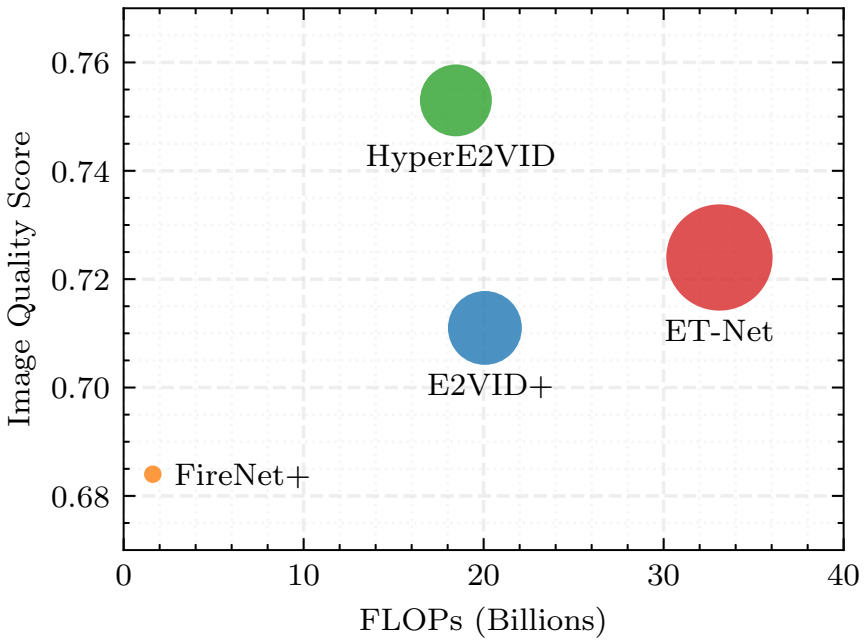


Figure 6.1 Comparison of our HyperE2VID method with state-of-the-art event-based video reconstruction methods based on image quality and computational complexity. Image quality scores are calculated by normalizing and averaging each of the quantitative scores reported in Table 7.9, where normalization maps the best and worst possible score for each metric to 1.0 and 0.0. The number of floating point operations (FLOPs) are measured as described in Section 7.7. Circle sizes indicate the number of model parameters, as detailed in Table 7.13. The methods with lower image quality scores are not included for clarity of presentation.

Figure 6.2 presents an overview of our proposed method, HyperE2VID, for reconstructing video from events. Our approach is designed to guide the dynamic filter generation through a *context* that represents the current scene being observed. To achieve this, we leverage two complementary sources of information: events and images. We incorporate a context fusion module in our hypernetwork architecture to combine information from event voxel grids and previously reconstructed intensity images. These two modalities complement each other since intensity images capture static parts of the scene better, while events excel at dynamic parts. By fusing them, we obtain a context tensor that better represents both static and dynamic parts of the scene. This tensor is then used to guide the dynamic per-pixel filter generation. We also employ a curriculum learning strategy to train the network more robustly, particularly in the early epochs of training when the reconstructed intensity images are far from optimal.

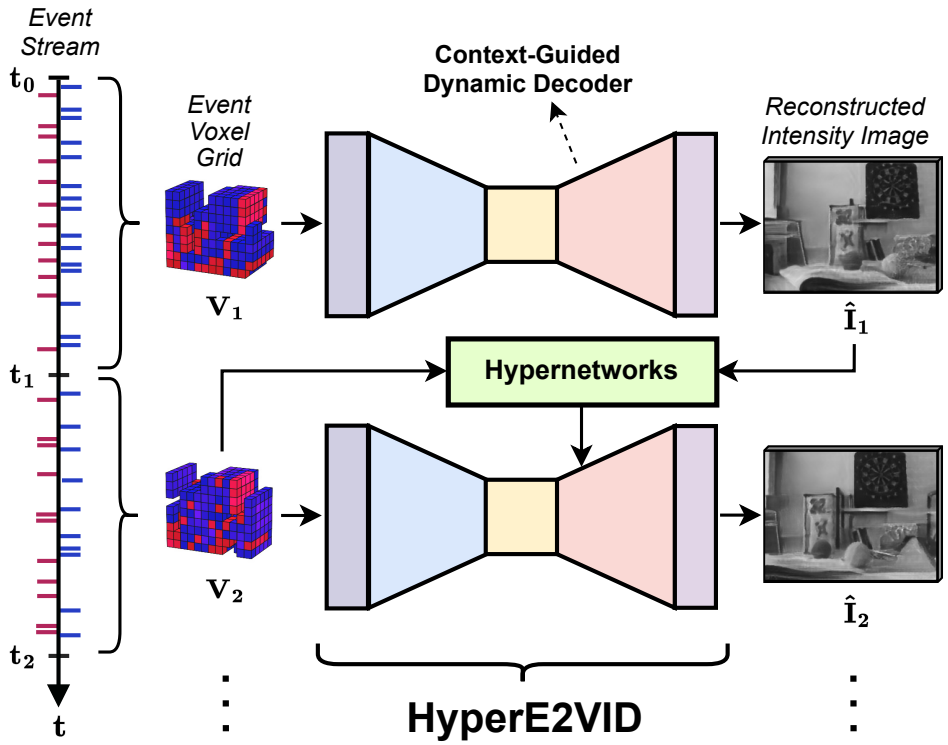


Figure 6.2 HyperE2VID uses a recurrent encoder-decoder backbone, consuming an event voxel grid at each time step. It enhances this architecture by employing per-pixel, spatially-varying dynamic convolutions at the decoder, whose parameters are generated dynamically at inference time via hypernetworks.

To the best of our knowledge, this is the first work that explores the use of hypernetworks and dynamic convolutions for event-based video reconstruction. The closest to our work is SPADE-E2VID [291] where the authors employ adaptive feature denormalization in decoder blocks of the E2VID architecture. Rather than feature denormalization, we directly generate per-pixel dynamic filters via hypernetworks for the first decoder block. Specifically, our contributions can be summarized as follows:

- We propose the first dynamic network architecture for the task of video reconstruction from events, where we extend existing static architectures with hypernetworks, dynamic convolutional layers, and a context fusion block.
- We show via experiments that this dynamic architecture can generate higher-quality videos than previous state-of-the-art, while also reducing memory consumption and inference time.

6.2. Dynamic Networks

Dynamic network is a generic term used to define a network that can adapt its parameters or computational graph dynamically according to its inputs at inference time [342]. This dynamic adaptation can be accomplished in many different ways. For example, one can use a hypernetwork [341], which is a smaller network that is used to dynamically generate weights of a larger network conditioned on the input. For convolutional networks, dynamic filter generation can be position specific as well, such that a different filter is generated for each spatial location and the filtering operation is not translation invariant anymore [344]. Position-specific dynamic filters can be pixel-wise, with a separate kernel for each spatial position, or patch-wise to reduce computational requirements. For example, Nirkin *et al*. proposed HyperSeg, a semantic segmentation network [345] where the encoder generates parameters for dynamic patch-wise convolutional layers in the decoder. In [346], Shaham *et al*. proposed a Spatially-Adaptive Pixel-wise Network (ASAP-Net), where a lightweight convolutional network acts as a hypernetwork. This hypernetwork works on a lower-resolution input and produces parameters of spatially varying pixel-wise MLPs that process each pixel of the higher-resolution input independently.

It is also possible to dynamically adjust network parameters rather than directly generating them, for example by applying soft attention over multiple convolutional kernels. Both Yang *et al*. [347] and Chen *et al*. [348] proposed to calculate a sample-specific convolutional kernel as a linear combination of many convolutional kernels, where combination coefficients are generated dynamically for each sample. Su *et al*. [349] introduced Pixel-Adaptive Convolution (PAC), where they modify the spatially invariant convolutional kernel by multiplying it with a spatially varying adapting kernel that depends on the input. Chen *et al*. [350] proposed to spatially divide the input feature into regions and process each region with a separate filter. Wang *et al*. [343] proposed Adaptive Convolutions with Dynamic Atoms (ACDA), where they generate sample-specific convolutional filters by multiplying pixel-wise dynamic filter atoms with learned static coefficients. They also decomposed the

dynamic atoms to reduce the computational requirements of calculating pixel-wise dynamic filters.

Another approach to dynamic filters is to adapt the shape of the convolutional kernel rather than its parameters. Deformable convolution [351] deforms the geometric structure of the convolutional filter to allow sampling from irregular points. This is achieved by augmenting each sampling location in the filter with dynamic offsets generated by another learned convolutional kernel.

6.2.1. Dynamic Networks for Event-Based Vision

Recently, the concept of dynamic networks have started to be used in event-based vision literature as well. In [352], [353] and [76], deformable convolution based feature alignment modules are used for event-based image reconstruction, super-resolution, and HDR imaging, respectively. Vitoria *et al*. [354] used modulated deformable convolutions for the task of event-based image deblurring, where event features encode the motion in the scene, in the form of kernel offsets and modulation masks. Xie *et al*. [355] employed dynamically updated graph CNN to extract discriminative spatio-temporal features for event stream classification.

While the aforementioned methods focus on dynamically changing the computational graphs of networks, there are also works that directly generate network parameters in a dynamic manner. For instance, in the task of event-based video super-resolution, Jing et al.[356] employed a network that takes event representations as inputs and generates parameters for dynamic convolutional layers. In contrast, we employ a context fusion mechanism and generate dynamic parameters guided by both event and image information, motivated by the complementary nature of these two domains. Xiao et al. [357] used dynamic convolutional filters similar to our method but for event-based video frame interpolation. However, they applied each convolutional kernel of shape $1 \times k \times k$ to a specific feature channel to reduce computational demand, which prevents effective modeling of inter-channel dependencies. On the other hand, we consider usual 2D convolutions to let the network model these dependencies, while avoiding high computational costs by using two filter decomposition

steps. Furthermore, we utilize previously reconstructed intensity images for context fusion and employ a curriculum learning strategy for robust training, as will be detailed later.

6.3. HyperE2VID

Here, we present details of our proposed method, HyperE2VID. Our task is to generate a stream of intensity images from a stream of events, as described formally in Section 5.2. Since each event conveys very little information regarding the scene, a common approach is to accumulate events into a group and then process this group together. We also follow this approach, using the *between-frames* grouping strategy described in Section 2.2.2., where we group events such that every event between consecutive frames ends up in the same group[5]. After event grouping, we utilize the voxel grid event representation presented in [201] and perform inference with our proposed neural network. The details of this event representation and our network architecture are described in the following sections.

6.3.1. Event Representation

Let G_k be a group of events that span a duration of ΔT seconds, T_k be the starting timestamp of that duration, and B be the number of temporal bins used to discretize the timestamps of continuous-time events in the group. The voxel grid $V_k \in \mathbb{R}^{W \times H \times B}$ for that group is formed by normalizing the timestamps of events from the group to the range $[0, B-1]$. Each event contributes its polarity to the two temporally closest voxels using a linearly weighted accumulation similar to bilinear interpolation. Specifically, the voxel grid is computed as follows:

$$V_k(x, y, t) = \sum_i p_i \max(0, 1 - |t - t_i^*|)\delta(x - x_i, y - y_i) \tag{7}$$

[5]This strategy assumes that the ground truth intensity frames are available with the incoming event stream. This assumption holds for the training phase since we use synthetic training data with ground truth frames, as described in Section 6.4.1. For the test time, the ground truth frames may or may not be available. When they are available, we again use *between-frames* event grouping strategy in our experiments; however, when they are not available, we adopt the *fixed-duration* event grouping strategy described in Section 2.2.2. The details of experimental setup, including these choices, are presented with more detail in Chapter 7.

where δ is the Kronecker delta that selects the pixel location, and t_i^* is the normalized timestamp which is calculated as:

$$t_i^* = (B-1)(t_i - T_k)/(\Delta T) \tag{8}$$

In all our experiments, we set the number of temporal bins B to 5.

6.3.2. Network Architecture

After representing each event group with a voxel grid, our task is to generate an image stream from the sequence of voxel grids. We use a recurrent neural network that consumes a voxel grid V_k at each time step $k \in \{1, \ldots, N_I\}$, and generates an image $\hat{I}_k$ corresponding to that specific moment. Specifically, we use a U-Net [300] based fully convolutional architecture with recurrent encoder blocks, decoder blocks, and skip connections between them, similar to the E2VID model [87] and the subsequent works of [12], [289], and [291]. Then, we augment this main architecture with hypernetworks, dynamic convolutions, and a context fusion module. We refer to the resulting architecture as HyperE2VID.

Figure 6.3 shows an overview of the proposed HyperE2VID framework. Our model consists of a main network $\mathcal{F}$ and hypernetworks that generate parameters for the dynamic part of the main network. From its input to output, the network $\mathcal{F}$ consists of one head layer, three recurrent encoder blocks, two residual blocks, one context-guided dynamic decoder (CGDD) block, two standard decoder blocks, and a prediction layer. The dynamic filter generation (DFG) block and the context fusion (CF) block act as hypernetworks that generate pixel-wise dynamic filter parameters for the dynamic part of the main network, *i.e.* the CGDD block.

More formally, let S_k be the recurrent state of the network for a time step k, containing states $\mathbf{S_k^{e_n}}$ of the three encoder blocks, where $n \in \{0, 1, 2\}$. Given the states from the previous time step, S_{k-1}, and the event voxel grid from the current time step, V_k, the main network $\mathcal{F}$

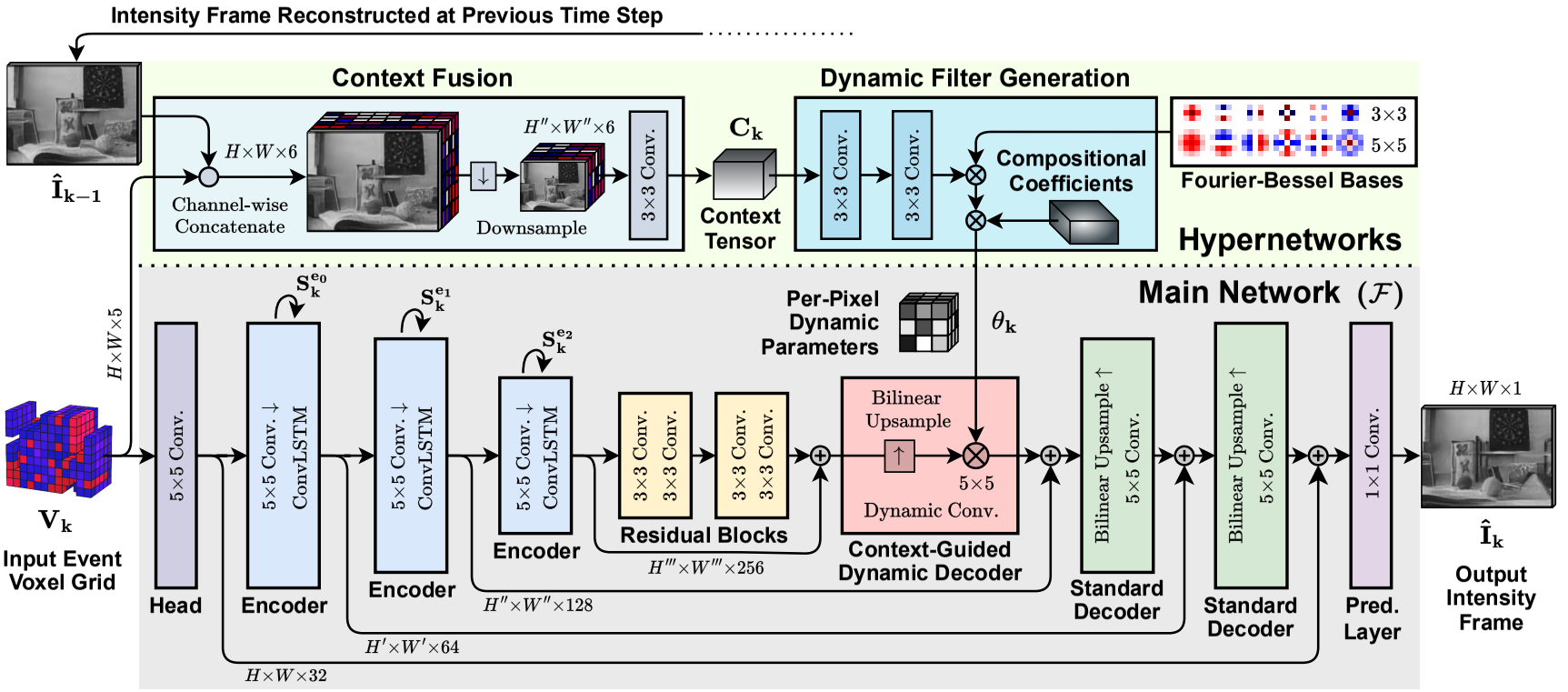


Figure 6.3 Overview of our proposed HyperE2VID architecture. The main network $\mathcal{F}$ uses a U-Net like architecture to process an event voxel grid V_k and predict the intensity image $\hat{I}_k$ at each time step k. It includes downsampling encoder blocks, upsampling decoder blocks, and skip connections. The encoders incorporate ConvLSTM blocks to capture long temporal dependencies in the sparse event stream. The parameters of the context-guided dynamic decoder (CGDD) block are generated dynamically at inference time, enabling the network to adapt to highly varying event data. These parameters are generated via hypernetworks, consisting of a context fusion (CF) block and a dynamic filter generation (DFG) block. The DFG block employs two filter decomposition steps using multi-scale Fourier-Bessel Bases and learned compositional coefficients, avoiding the high computational cost of per-pixel adaptive filters. The CF block fuses event features from the current time step k with reconstructed image features from the previous time step $k-1$ to generate a context tensor. This fusion scheme combines the dynamic and static parts of the scene captured by events and images, respectively, to generate a context tensor that better represents the overall scene.

calculates the current states S_k and predicts the intensity image $\hat{I}_k$ as follows:

$$(\hat{I}_k, S_k) = \mathcal{F}(V_k, S_{k-1}, \theta_k) \tag{9}$$

with θ_k denoting the parameters of the convolutional layer at the CGDD block, which are generated dynamically at inference time by the DFG block, as below:

$$C_k = \text{CF}(V_k, \hat{I}_{k-1}) \tag{10}$$

$$\theta_k = \text{DFG}(C_k) \tag{11}$$

To generate the parameters of the dynamic decoder, we use both the current event voxel grid V_k and the previous reconstruction result $\hat{I}_{k-1}$. The CF block fuses these inputs to generate a context tensor C_k, which is then used by the DFG block. This approach is motivated by the complementary nature of the two domains. Events are better suited for capturing fast motion due to their high temporal resolution but cannot capture static parts of the scene. In contrast, intensity images are better at capturing static parts of the scene. By fusing V_k and $\hat{I}_{k-1}$, the context tensor C_k incorporates useful features that better describe the static and dynamic parts of the scene.

Skip connections carry output feature maps of the head layer and each encoder block to the inputs of the respective symmetric decoder components, *i.e*. before each decoder block and the prediction layer. Element-wise summation is performed for these skip connections. ReLU activations are used for each convolutional layer unless specified otherwise. We describe each component of our architecture in more detail below:

Head layer. The head layer consists of a convolutional layer with a kernel size of 5. The convolutional layer processes the event voxel grid with 5 temporal channels and outputs a tensor with 32 channels, while the input's spatial dimensions H and W are maintained.

Encoder blocks. Each encoder block consists of a convolutional layer followed by a ConvLSTM [301]. The convolutional layer has a kernel size of 5 and stride of 2, thus, it

reduces the spatial dimensions of the input feature map by half. On the other hand, it doubles the number of channels. The ConvLSTM has a kernel size of 3 and maintains the spatial and channel dimensions of its inputs and internal states.

Residual blocks. Each residual block in our network comprises two convolutional layers with a kernel size of 3 that preserve the input's spatial and channel dimensions. A skip connection adds the input features to the output features of the second convolution before the activation function.

Context-Guided Dynamic Decoder (CGDD) block. The CGDD block includes bilinear upsampling to increase the spatial dimensions, followed by a dynamic convolutional layer. The convolution contains 5×5 kernels and reduces the channel size by half. The parameters θ_k of this convolution are generated dynamically during inference time by the DFG block.

It is important to emphasize that all dynamic parameters are generated pixel-wise in that there exists a separate convolutional kernel for each pixel. This spatial adaptation is motivated by the fact that the pixels of an event camera work independently from each other. When there is more motion in one part of the scene, events are generated at a higher rate at corresponding pixels, and the resulting voxel grid is denser in those regions. Our design enables the network to learn and use different filters for each part of the scene according to different motion patterns and event rates, making it more effective to process the event voxel grid with spatially varying densities.

Standard Decoder blocks. Each standard decoder block consists of bilinear upsampling followed by a standard convolutional layer. The details are the same as the context-guided dynamic decoder, except that the parameters are learned at training time and fixed at inference time. As part of our ablation studies (Section 7.8.), we also explore an architectural variant where we employ sub-pixel convolutions [308] instead of bilinear upsampling in decoder blocks.

Prediction layer. The prediction layer is a standard convolutional layer with a kernel size of 1, and it outputs the final predicted intensity image with 1 channel. We do not use an

activation function after this layer.

Dynamic Filter Generation (DFG) block. A crucial component of our method is the dynamic filter generation. This block consumes a *context* tensor and output parameters for the CGDD block. The context tensor C_k is expected to be at the same spatial size as the input of the dynamic convolution ($W'' \times H''$). To generate the context tensor, we use a context fusion mechanism that fuses features from the event voxel grid (V_k) and the previous reconstruction ($\hat{I}_{k-1}$) of the network.

To reduce the computational cost, we use two filter decomposition steps while generating per-pixel dynamic filters. First, we decompose filters into per-pixel filter atoms generated dynamically. Second, we further decompose each filter atom as a truncated expansion with pre-fixed multi-scale Fourier-Bessel bases. Inspired by ACDA [343], our approach generates efficient per-pixel dynamic convolutions that vary spatially. However, unlike ACDA, our network architecture performs dynamic parameter generation independently through hypernetworks, which are guided by a context tensor designed to provide task-specific features for event-based video reconstruction.

Figure 6.4 illustrates the detailed operations of our proposed DFG block. A context tensor with dimensions $W'' \times H'' \times C_{cont}$ is fed into a 2-layer CNN, producing pixel-wise basis coefficients of size C_{coeff} that are used to generate per-pixel dynamic atoms via pre-fixed multi-scale Fourier-Bessel bases. These bases are represented by a tensor of size $s \times b \times l \times l$, where s is the number of scales, b is the number of Fourier-Bessel bases at each scale, and l is the kernel size for which the dynamic parameters are being generated. Multiplying the multi-scale Fourier-Bessel bases with the basis coefficients generate per-pixel dynamic atoms of size $l \times l$. Number of generated atoms for each pixel is a, so it is possible to represent all of the generated atoms by a tensor of size $W'' \times H'' \times a \times l \times l$. Next, the compositional coefficients tensor of size $C_{in} \times a \times C_{out}$ is multiplied with these per-pixel dynamic atoms. These learned coefficients are fixed at inference time and shared across spatial positions. This multiplication produces a tensor of size $W' \times H' \times C_{in} \times C_{out} \times l \times l$, which serves as the parameters for the per-pixel dynamic convolution. Here, C_{in} and C_{out}

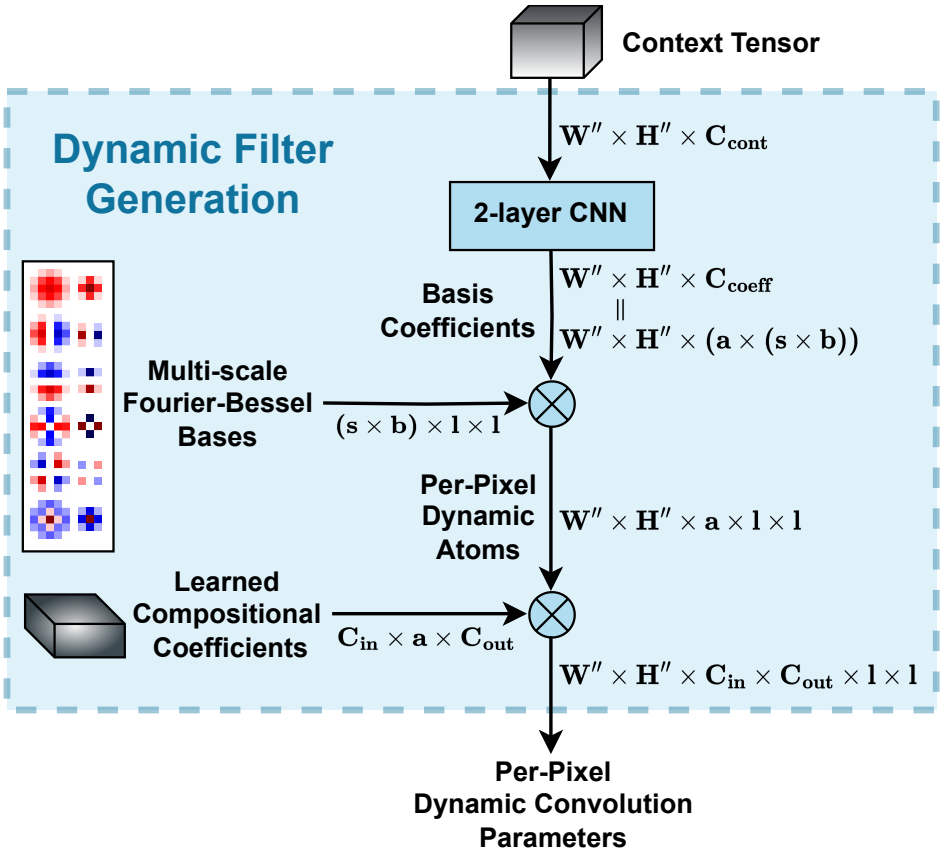


Figure 6.4 Dynamic Filter Generation (DFG) block. DFG block takes a context tensor as input and generates per-pixel dynamic convolution parameters via two filter decomposition steps, making use of pre-fixed multi-scale Fourier-Bessel bases and learned compositional coefficients. More details are given in Section 6.3.2.

are the number of input and output channels for the dynamic convolution, respectively. For the DFG block, we set $a = 6$, $b = 6$, and $l = 5$. The number of scales $s = 2$, meaning that we use 3×3 and 5×5 sized Fourier-Bessel bases. Since we have $b = 6$ bases at each scale, we have a total of $s \times b = 12$ Fourier-Bessel bases. The 2-layer CNN has a hidden channel size of 64. Both convolutional layers have a kernel size of 3, and they are followed by a batch normalization [358] layer and a tanh activation. The output of the CNN has $C_{coeff} = a \times b \times s = 72$ channels, to produce a separate coefficient per dynamic atom and per Fourier-Bessel basis.

Context Fusion (CF) block. The events are generated asynchronously only when the intensity of a pixel changes, and therefore the resulting event voxel grid is a sparse tensor, incorporating information only from the changing parts of the scene. Our HyperE2VID

architecture conditions the dynamic decoder block parameters with both the current event voxel grid V_k and the previous network reconstruction $\hat{I}_{k-1}$. These two domains provide complementary information; the intensity image is better suited for static parts of the scene, while the events are better for dynamic parts. We use the CF block to fuse this information, enabling the network to focus on intensity images for static parts and events for dynamic parts. Our context fusion block design concatenates V_k and $\hat{I}_{k-1}$ channel-wise to form a 6-channel tensor. We downsample this tensor to match the input dimensions of the dynamic convolution at the CGDD block and then use a 3×3 convolution to produce a context tensor with 32 channels. While more complex architectures are possible, we opt for a simple design for the context fusion block.

6.4. Training Details

6.4.1. Training Data

We generate a synthetic training set as described in [289], using the `Multi-Objects-2D` renderer option of ESIM [294] where multiple moving objects are captured with a camera restricted to 2D motion. The dataset consists of 280 sequences, all of which are 10 secs in length. The contrast threshold values for event generation are in the range of 0.1 to 1.5. Each sequence includes generated event streams together with ground truth intensity images and optical flow maps with an average rate of 51 Hz. The resolutions of event and frame cameras are both 256×256. The sequences include scenes containing up to 30 foreground objects with varying speeds and trajectories, where the objects are randomly selected images from the MS-COCO dataset [114].

Data Augmentation. During training, we augment the images and event tensors with random crops and flips as suggested in [12]. The size of random crops is 112×112, and the probability of vertical and horizontal flips are both 0.5. Furthermore, we employ dynamic train-time noise augmentation, pause augmentation, and hot-pixel augmentation as described in [289].

6.4.2. Loss Functions

During training, we employ the following loss functions:

Perceptual Reconstruction Loss. We use the AlexNet [13] variant of the learned perceptual image patch similarity (LPIPS) [2] to enforce reconstructed images to be perceptually close to ground truth intensity images. LPIPS works by passing the predicted and reference images through a deep neural network architecture that was trained for visual recognition tasks, and using the distance between deep features from multiple layers of that network as a measure of the perceptual difference between the two images.

$$\mathcal{L}_k^{\text{LPIPS}} = \text{LPIPS}(\hat{I}_k, I_k) \tag{12}$$

Temporal Consistency Loss. We use the *short-term temporal loss* of [3], as employed in [12], to enforce temporal consistency between the images that are reconstructed in consecutive time steps of the network. This loss works by warping the previously reconstructed image using a ground truth optical flow to align it with the current reconstruction and using a masked distance between these aligned images as a measure of temporal consistency, where the mask is calculated from the warping error between the previous and the current ground truth intensity images. More formally, the temporal consistency loss is calculated as:

$$\mathcal{L}_k^{\text{TC}} = M_k \| \hat{I}_k - W(\hat{I}_{k-1}, F_{k \to k-1}) \|_1 \tag{13}$$

where $F_{k \to k-1}$ denotes the optical flow map between time steps k and $k-1$, W is the warping function, and M_k represents the occlusion mask which is computed as:

$$M_k = \exp(-\alpha \| I_k - W(I_{k-1}, F_{k \to k-1}) \|_2^2) \tag{14}$$

where we use $\alpha = 50$ as in [3, 12]. The mask M_k contains smaller terms for pixels where the warping error between consecutive ground truth images is high, and therefore the

masking operation effectively discards these pixels from the temporal consistency calculation of reconstructed frames.

The final loss for a time step k is the sum of the perceptual reconstruction and temporal losses:

$$\mathcal{L}_k = \mathcal{L}_k^{\text{LPIPS}} + \mathcal{L}_k^{\text{TC}} \tag{15}$$

During training, we calculate the loss $\mathcal{L}_k$ at every T_S time-steps in a training sequence, and the gradients of this loss with respect to the network parameters are calculated using the Truncated Back-propagation Through Time (TBPTT) algorithm [359] with a truncation period of T_T time-steps. Setting $T_S > 1$ and $T_T < k$ reduces memory requirements and speeds up the training process.

Our choice of perceptual and temporal losses over traditional content consistency losses like L1 or L2 is motivated by the inadequacy of per-pixel losses (such as L1 and L2) in capturing human visual perception. These traditional losses, focusing on pixel-wise accuracy, often prove inadequate when evaluating structured outputs like images since they assume pixel-wise independence, and fail to account for perceptual similarities, as evidenced by Zhang et al. [2]. They can also lead to blurry images in reconstruction tasks, as highlighted by Johnson et al. [360]. Moreover, Blau and Michaeli [361] have shown a trade-off between distortion and perceptual quality in image generation. Given our goal of reconstructing high-quality intensity images for better human interpretation and enabling the application of frame-based computer vision methods to event data, achieving high perceptual quality is important. Thus, we opt for perceptual and temporal losses, which align more closely with human visual perception and are more suited for mid to high-level computer vision tasks. To validate our approach, we conduct ablation experiments comparing the performance of our method with and without various loss functions. The details and results of these ablation experiments are given in Section 7.8.

6.4.3. Curriculum Learning

At the start of the training, the previous reconstruction $\hat{I}_{k-1}$ of the network, which is used for context fusion, is far from optimal. This makes it harder for the context fusion block to learn useful representations, especially in the earlier epochs of the training. To resolve this issue, we employ a curriculum learning [362] strategy during the training. We start the training by using the ground truth previous image I_{k-1} instead of the previous reconstruction of the network $\hat{I}_{k-1}$ for context fusion. For the first 100 epochs, we gradually switch to using images that the network reconstructs at the previous time step, by weighted averaging them with ground-truth images. After the 100th epoch, we continue the training by only using previous reconstructions for context fusion. Therefore, we use a modified version of Equation (10) during training:

$$\beta = \min(1,\ \frac{\mathit{epoch}}{100}) \tag{16}$$

$$I_{\mathit{context}} = \beta \cdot \hat{I}_{k-1} + (1 - \beta) \cdot I_{k-1} \tag{17}$$

$$C_k = \mathrm{CF}(V_k, I_{\mathit{context}}) \tag{18}$$

This curriculum learning strategy allows the parameters of the hypernetworks to be learned more robustly, enabling the training process to converge to a better-performing model.

6.4.4. Hyperparameters and Implementation Details

We implement our network in PyTorch [333]. We train the recurrent network with sequences of length 40, with the network parameters initialized using He initialization [363]. At the first time step of each sequence, the initial values of the previous reconstruction, $\hat{I}_0$, and the network states, S_0, are set to zero tensors. The loss calculation and the truncation periods are set as $T_S = 10$ and $T_T = 5$, respectively. We train our network for 400 epochs using a batch size of 10 and the AMSGrad [364] variant of the Adam [365] optimizer with a learning rate of 0.001. To track our trainings and experimental analyses, we use Weights & Biases [366].

7. EXPERIMENTAL RESULTS

This chapter details our experimental efforts, outlining our setup, presenting extensive results, and engaging in discussions about them. It delineates the sequences used in our experiments and the image quality metrics applied. Here, we assess HyperE2VID and other methods across multiple datasets, using both full-reference and no-reference image quality metrics and considering various downstream tasks. Additionally, we explore model robustness and computational complexity and conduct a comprehensive ablation study to evaluate the influence of different design aspects of HyperE2VID.

The experimental setup and results that we present in this chapter are a significantly expanded version of those in our published work, [332] and [334]. Specifically, we provide more details on the sequences by presenting their exact cuts (start and end times) in several tables in Section 7.1.1. We also include an analysis of underexposed reference frames in benchmark datasets and the effects of applying histogram equalization on them, detailed in Section 7.1.3. Moreover, this chapter presents additional quantitative and qualitative results that were not included in our previous work. Perhaps the most notable addition is the experiments using our proposed dataset HUE, featuring both quantitative and qualitative results. Furthermore, we include results from downstream tasks and additional ablation studies for HyperE2VID, which were not available in [334]. Finally, we present an alternative HyperE2VID architecture at the end of this chapter, which incorporates modified upsampling blocks to mitigate checkerboard artifacts.

7.1. Experimental Setup

We use our proposed evaluation framework EVREAL as the basis of our experimental setup. EVREAL enables us to evaluate and compare various methods from the literature using several datasets and via full-reference and no-reference image quality metrics. The datasets we employ are detailed in Section 7.1.1., while Section 7.1.2. presents the implementation details of image quality metrics that we use.

When ground truth frames are available, we employ *between-frames* event grouping, match each reconstruction with a ground truth frame, and evaluate reconstruction quality using full-reference evaluation metrics. On the other hand, when the benchmark datasets do not include high-quality ground truth frames, we use *fixed-duration* event grouping with a duration of $40\,\mathrm{ms}$ and assess reconstruction quality by using no-reference metrics. For all experiments, we use the voxel-grid event representation presented in Section 6.3.1.

As described in Section 5.4., EVREAL supports eight methods from the literature that have PyTorch-based open-source model codes and pre-trained models, including our proposed method HyperE2VID. In our experiments, we compare all these eight methods. For E2VID, FireNet, and SSL-E2VID, we normalize event voxel grids as suggested in these methods. Furthermore, we perform robust min/max normalization as a post-processing step for E2VID and SSL-E2VID. For SSL-E2VID, we also apply the exponential function before this min/max normalization.

7.1.1. Datasets

To comprehensively evaluate our method and compare it with other methods from the literature, we utilize sequences from our proposed dataset HUE (Chapter 4.), as well as eight other real-world datasets from the literature, each selected for its unique characteristics and relevance to different aspects of event-based video reconstruction. These datasets are the Event Camera Dataset (ECD) [280], the Multi Vehicle Stereo Event Camera (MVSEC) dataset [322], the High-Quality Frames (HQF) dataset [289], the Beam Splitter Event and RGB (BS-ERGB) Dataset [327], the HDR dataset [12], the UZH-FPV Drone Racing (FPVDR) dataset [328], the Color Event Camera Dataset (CED) [367], and the Neuromorphic-Caltech101 (N-Caltech101) dataset [318].

The HUE, ECD, MVSEC, HQF, BS-ERGB, HDR, and FPVDR datasets are mainly used to evaluate image quality quantitatively and quantitatively, as described in Section 7.2. Here, we utilize full-reference metrics to evaluate the image quality of each method when the datasets include high-quality ground truth frames. Furthermore, we use no-reference metrics to assess

the performance of the methods in challenging scenarios with fast motion, low light, and high dynamic range scenes since reference frames are of low quality in these scenarios or do not exist at all. The metrics we use are given in Section 7.1.2. with more details. The seventh dataset, CED, is used to qualitatively demonstrate the color reconstruction performance of HyperE2VID alongside other competing approaches. We generate color reconstructions as described in Section 5.7., and the results are presented in Section 7.3. Finally, N-Caltech101 is used to assess reconstruction performance via a downstream task, image classification, as explained in Section 5.8. The results of all downstream tasks are given in Section 7.6. In the following paragraphs, we detail how we utilize each dataset.

With frames and events generated from a DAVIS240C sensor, the ECD dataset is pivotal for evaluating reconstructions in indoor environments. Following the common practice established by Rebecq *et al*. [12], we use seven short sequences from this dataset. These sequences mostly contain simple office environments with static objects, and the camera moves with 6-DOF and increasing speed. Following [12], we exclude the initial few seconds of each sequence from quantitative evaluation to allow methods to generate meaningful results. Additionally, when using full-reference metrics, as commonly done in earlier work, we do not include the latter parts of the sequences as they may contain motion blur due to the increased speed of camera movement. Table 7.1 presents these seven sequences, the exact start and end times of quantitative evaluation with full-reference metrics, and the total number of evaluated frames. These sequences provide ground truth intensity frames at an average rate of $22\,\mathrm{Hz}$, and we employ the *between-frames* event grouping strategy, *i.e*. we group events with timestamps between each consecutive ground truth frame together.

Furthermore, we introduce a subset of the ECD, which we denote as ECD-Fast. This subset contains the latter parts of ECD sequences where the camera undergoes fast motion. With ECD-Fast, we utilize no-reference image quality metrics to assess reconstruction quality under rapid camera motion. The sequences in this subset are given in Table 7.2, together with specific start and end times for no-reference image quality assessment.

The MVSEC dataset offers longer sequences captured by a DAVIS 346B camera in

Table 7.1 The sequences used from the ECD dataset for quantitative evaluation with the full-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
boxes_6dof	5.0	20.0	326
calibration	5.0	20.0	357
dynamic_6dof	5.0	20.0	318
office_zigzag	5.0	12.0	133
poster_6dof	5.0	20.0	340
shapes_6dof	5.0	20.0	340
slider_depth	1.0	2.5	39
Total			1853

Table 7.2 The sequences used from the ECD-Fast dataset for quantitative evaluation with the no-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
boxes_6dof	20.0	59.8	995
calibration	20.0	59.8	996
dynamic_6dof	20.0	59.8	994
poster_6dof	20.0	59.8	995
shapes_6dof	20.0	59.8	993
Total			4973

indoor and outdoor settings. This dataset is integral for analyzing performance in diverse environments. The original dataset includes a stereo DAVIS camera setup, and we use the data from the left DAVIS camera in our experiments. Following [289], we use specific time intervals of six sequences. Four of them are indoor sequences taken from a flying hexacopter, while the two outdoor sequences are taken from a vehicle driving in daylight. The average rate of ground truth intensity frames is around $30\,\mathrm{Hz}$ for indoor sequences and $45\,\mathrm{Hz}$ for outdoor sequences. The specific time intervals and total number of evaluated frames are given in Table 7.3.

Table 7.3 The sequences used from the MVSEC dataset for quantitative evaluation with the full-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
indoor_flying1_data	10.0	70.0	1884
indoor_flying2_data	10.0	70.0	1884
indoor_flying3_data	10.0	70.0	1884
indoor_flying4_data	10.0	19.8	308
outdoor_day1_data	0.0	60.0	2740
outdoor_day2_data	100.0	160.0	2625
Total			11325

Table 7.4 The sequences used from the MVSEC-Night dataset for quantitative evaluation with the no-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
outdoor_night1_data	0.0	262.1	6551
outdoor_night2_data	0.0	374.4	9360
outdoor_night3_data	0.0	276.7	6918
Total			22829

Additionally, we derive the MVSEC-Night subset to evaluate our method's effectiveness in low-light conditions, a challenging scenario for event-based reconstruction. We use three night driving sequences from the MVSEC dataset and employ no-reference image quality metrics to assess reconstruction quality under low light. The details of these sequences are given in Table 7.4.

The HQF dataset provides a variety of indoor and outdoor sequences with well-exposed and minimally blurred frames, providing benchmarking in more controlled environments. The dataset offers ground truth intensity frames with an average rate of $22.5\,\mathrm{Hz}$, and following [289], we use the entire sequences from this dataset for evaluation using *between-frames* event grouping and full-reference quantitative metrics (see Table 7.5).

Table 7.5 The sequences used from the HQF dataset for quantitative evaluation with the full-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
bike_bay_hdr	0.0	98.9	2430
boxes	0.0	24.1	539
desk	0.0	65.6	1490
desk_fast	0.0	31.8	723
desk_hand_only	0.0	20.5	466
desk_slow	0.0	63.1	1433
engineering_posters	0.0	60.6	1266
high_texture_plants	0.0	43.1	1089
poster_pillar_1	0.0	41.7	997
poster_pillar_2	0.0	25.3	612
reflective_materials	0.0	28.8	618
slow_and_fast_desk	0.0	75.5	1743
slow_hand	0.0	38.8	900
still_life	0.0	67.9	1193
Total			15499

We also created the HQF-Slow subset within HQF to test our method's performance in slow-motion scenes, which present unique challenges due to reduced event rates. This subset includes all 2333 ground truth frames from two sequences named `desk_slow` and `slow_hand`, which were collected with the explicit aim of incorporating slow-motion scenarios. We utilize this subset in our ablation studies presented in Section 7.8., specifically to assess the effect of context information.

Among these datasets, the BS-ERGB dataset has the highest resolution, making it an important addition for assessing the performance of each method when reconstructing larger frames. Although the dataset is originally captured with event and frame sensors of resolutions 1280×720 pixels and 4096×2196 pixels, these events and frames are then post-processed to match each other spatially, and the final dataset has events and frames with the spatial resolution of 970×625 pixels. Most of the sequences in this dataset are short and

Table 7.6 The sequences used from the BS-ERGB dataset for quantitative evaluation with the full-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
may29_handheld_01	0.0	12.4	350
may29_handheld_02	0.0	7.85	222
may29_handheld_03	0.0	16.0	450
may29_handheld_04	0.0	5.7	160
may29_rooftop_handheld_01	0.0	28.7	592
may29_rooftop_handheld_02	0.0	20.7	427
may29_rooftop_handheld_03	0.0	17.8	367
may29_rooftop_handheld_05	0.0	12.9	267
street_crossing_07	0.0	43.5	1226
street_crossing_08	0.0	18.8	530
Total			4591

captured with a static camera observing fast motions in the scene. In these sequences, events are confined to small regions where motion is observed, and reconstructing intensity frames for other parts of the scene is not feasible. However, a few other sequences are recorded with a handheld camera where every pixel generates many events, so we use these sequences in our evaluations. These ten handheld sequences and their details are presented in Table 7.6. We use *between-frames* event grouping and full-reference metrics to assess image quality with these sequences.

We use all three HDR sequences from [12], whose details are given in Table 7.7. For these sequences, we form event groups that span $40\,\mathrm{ms}$ and employ no-reference image quality metrics.

The UZH-FPV Drone Racing (FPVDR) dataset is captured by a mDAVIS346 camera mounted on a quadcopter flown by an expert drone racing pilot with fast and aggressive movements. The dataset consists of 26 indoor and outdoor flight sequences, with a total flight distance of more than $10\,\mathrm{km}$. The events and frames are generated from the same 346×260 pixel array of mDAVIS, which is positioned either forward facing or 45-degree

Table 7.7 The sequences used from the HDR dataset for quantitative evaluation with the no-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
hdr_selfie	0.0	5.1	129
hdr_sun	0.0	7.8	196
hdr_tunnel	0.0	11.2	280
Total			605

downward facing for each flight. With its fast and aggressive drone movements, FPVDR is ideal for testing our method under extreme motion conditions, offering a rigorous assessment of reconstruction capabilities in dynamic scenarios. We use no-reference metrics in these challenging scenarios since the reference frames mostly contain motion blur. We exclude the first few seconds of each sequence to start quantitative evaluation after the drone takes off. We use event groups spanning $40\,\mathrm{ms}$ and evaluate 24575 reconstructed frames.

The CED dataset's color frames and events, captured with the Color-DAVIS346 camera, allow us to demonstrate our method's color reconstruction ability. In particular, we use a few sequences with vibrant colors and challenging lighting conditions to present visual results of color reconstructions. The results are given in Section 7.3.

7.1.2. Image Quality Metrics

As explained in Section 5.5. and Section 7.1.1., we use both full-reference and no-reference metrics to assess image quality. Full-reference metrics are used when the datasets include high-quality ground truth frames, while no-reference metrics are used when the reference frames are of low quality or do not exist at all. In our experiments, we utilize three full-reference metrics, MSE, SSIM [1], and LPIPS [2]; and three no-reference metrics, BRISQUE [4], NIQE [5], and MANIQA [6]. The implementation details and specific settings of these metrics are presented below to ensure consistency and aid reproducibility:

Table 7.8 The sequences used from the FPVDR dataset for quantitative evaluation with the no-reference metrics. For each sequence, we denote the start and end times of quantitative evaluation in seconds, as well as the total number of evaluated frames.

Sequence	Evaluation Start [s]	Evaluation End [s]	Frames Evaluated
indoor_45_1	10.0	73.0	300
indoor_45_2	10.0	55.0	1025
indoor_45_3	10.0	57.0	800
indoor_45_4	10.0	47.0	825
indoor_45_9	10.0	40.0	125
indoor_45_11	10.0	22.0	1575
indoor_45_12	10.0	51.0	1125
indoor_45_13	10.0	42.0	1175
indoor_45_14	10.0	43.0	925
indoor_45_16	10.0	15.0	750
indoor_forward_3	10.0	54.0	575
indoor_forward_5	10.0	50.0	350
indoor_forward_6	10.0	32.0	525
indoor_forward_7	10.0	73.0	1100
indoor_forward_8	10.0	132.0	1000
indoor_forward_9	10.0	34.0	550
indoor_forward_10	10.0	33.0	1575
indoor_forward_11	10.0	24.0	3050
indoor_forward_12	10.0	31.0	600
outdoor_forward_1	10.0	49.0	1225
outdoor_forward_2	10.0	36.0	975
outdoor_forward_3	10.0	92.0	650
outdoor_forward_5	10.0	22.0	2050
outdoor_forward_6	10.0	34.0	300
outdoor_forward_9	10.0	43.0	600
outdoor_forward_10	10.0	59.0	825
Total			24575

MSE. The Mean Squared Error is a commonly used metric that does not require any parameters. When comparing two images, the only factor that can impact the MSE result is the range of pixel values that the images possess. We calculate the MSE using floating-point pixel values within the range $[0, 1]$. Lower MSE values indicate better results.

SSIM. We utilize the `scikit-image` image processing library's [368] implementation for structural similarity. We adjust the parameters to use the Gaussian weighting scheme explained in [1]. Like MSE, we compute SSIM using images with floating point pixel values in the range of $[0, 1]$. Higher scores of SSIM indicate better results.

LPIPS. For LPIPS, we use the implementation in IQA-PyTorch toolbox [335][6], v0.1.10. We employ the default settings, using the LPIPS variant that uses the pre-trained AlexNet [13] network. The implementation supports 3-channel RGB images. Therefore, we convert intensity images into RGB images by concatenating three copies of the grayscale image along the channel axis before calculating the scores. In the LPIPS score calculation, a lower score indicates better quality.

BRISQUE. For BRISQUE [4], we again use the implementation in IQA-PyTorch toolbox [335], v0.1.10, with default settings. Similarly, we concatenate three copies of the grayscale image along the channel axis. Lower BRISQUE scores are better.

NIQE. For NIQE [5], we again use the implementation in IQA-PyTorch toolbox [335], v0.1.10, with default settings, and concatenating three copies of the grayscale image along the third dimension. Lower NIQE scores are better.

MANIQA. For MANIQA [6], we follow the same approach and use the implementation in IQA-PyTorch toolbox [335], v0.1.10, with default settings. In contrast to the above metrics, higher MANIQA scores imply higher image quality.

7.1.3. Analysis on Darker Ground Truth Frames and Histogram Equalization

The authors of E2VID [12] incorporate local histogram equalization as part of their evaluation procedure, applying it to both ground-truth and reconstructed frames before calculating quantitative image quality metrics. This approach may be in response to lower

[6]The code is accessible from `https://github.com/chaofengc/IQA-PyTorch`

intensity values present in some ground-truth images—a prevalent issue across the ECD, MVSEC, and HQF datasets, commonly used for evaluating event-based video reconstruction. However, addressing this challenge is not straightforward. Here, we perform an analysis of these underexposed frames, examine the effects of applying histogram equalization on them, and explain our reasons for choosing not to apply histogram equalization.

We identify two distinct scenarios of underexposure: one where parts of a ground truth image exhibit zero intensity due to severe underexposure, resulting in a loss of visual information, and another one where an image appears dark overall due to mild underexposure without necessarily losing information. The ECD and MVSEC datasets, which are captured with DAVIS240C and DAVIS 346B cameras, exhibit both of these issues, with the former being more pronounced in ECD. The HQF dataset, designed specifically for event-based video reconstruction, generally has better exposure but still faces the issue of zero-intensity regions.

We should note that recovery of visual information from zero-intensity regions is impossible using methods like histogram equalization; hence, the first issue is unsolvable through these means. While the second issue can be somewhat mitigated, the methods often introduce artifacts such as amplified noise and result in unrealistic image appearance.

Consider the sample scenes from ECD, MVSEC, and HQF datasets given in Figures 7.1, 7.2, and 7.3, respectively. In these figures, the leftmost column displays the ground truth frame for each scene, including both original and processed versions. The second column highlights regions with zero intensity values, indicative of lost visual information. The third and fourth columns depict histograms of image intensities, including and excluding zero-intensity pixels, respectively. The latter histogram aims to enhance the clarity of the intensity distribution among the remaining (non-zero) pixels. In each figure, the top row shows the original dataset image, while the others show variations processed through global histogram equalization and local histogram equalization.

These figures reveal that while global histogram equalization improves contrast, it cannot address the zero-intensity regions. Consequently, the binary images for these processed versions remain unchanged. Additionally, it tends to increase noise alongside contrast.

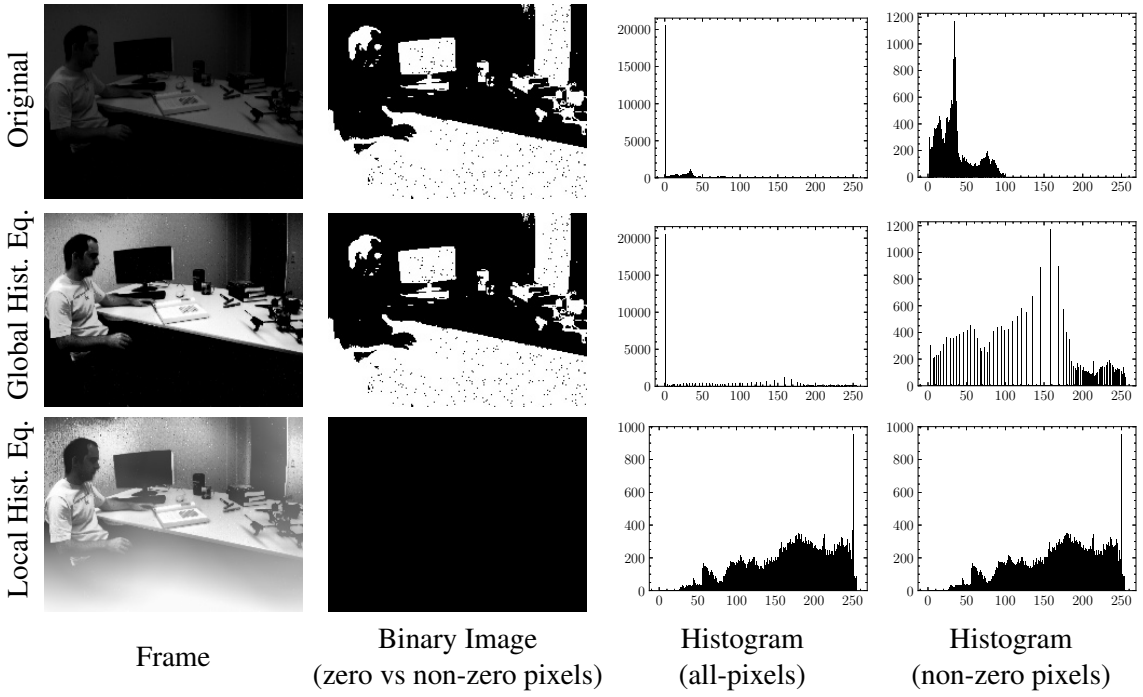


Figure 7.1 An underexposed frame from the ECD dataset, showing the ground truth frame and the corresponding zero intensity areas, together with the effects of global and local histogram equalization.

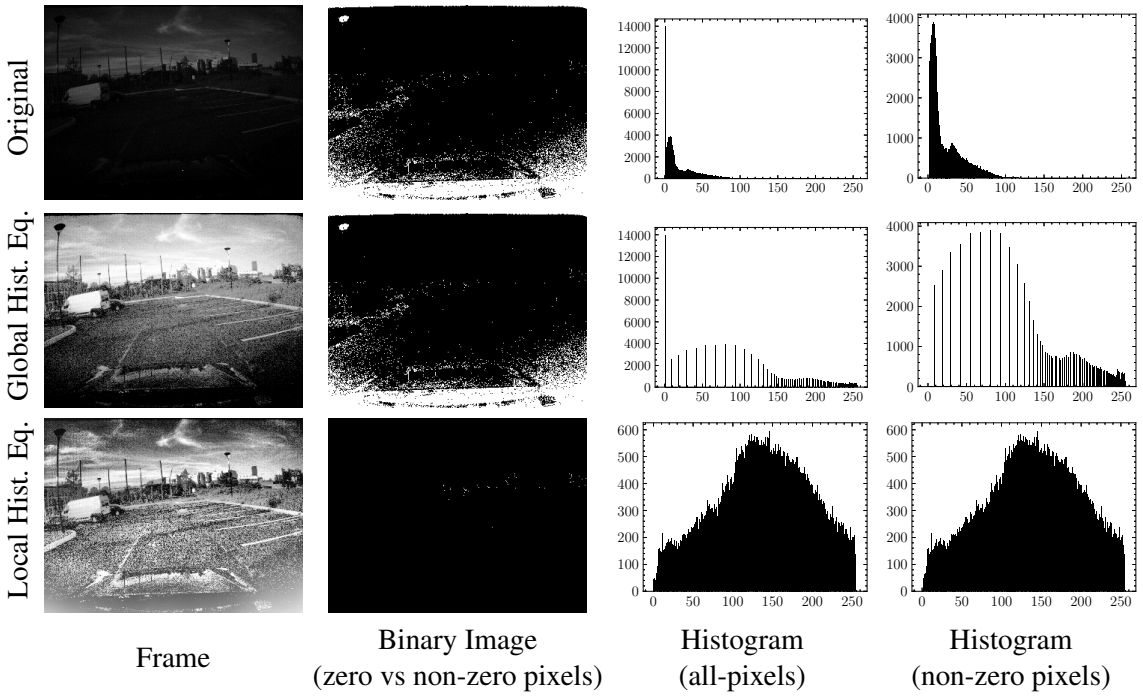


Figure 7.2 An underexposed frame from the MVSEC dataset, showing the ground truth frame and the corresponding zero intensity areas, together with the effects of global and local histogram equalization.

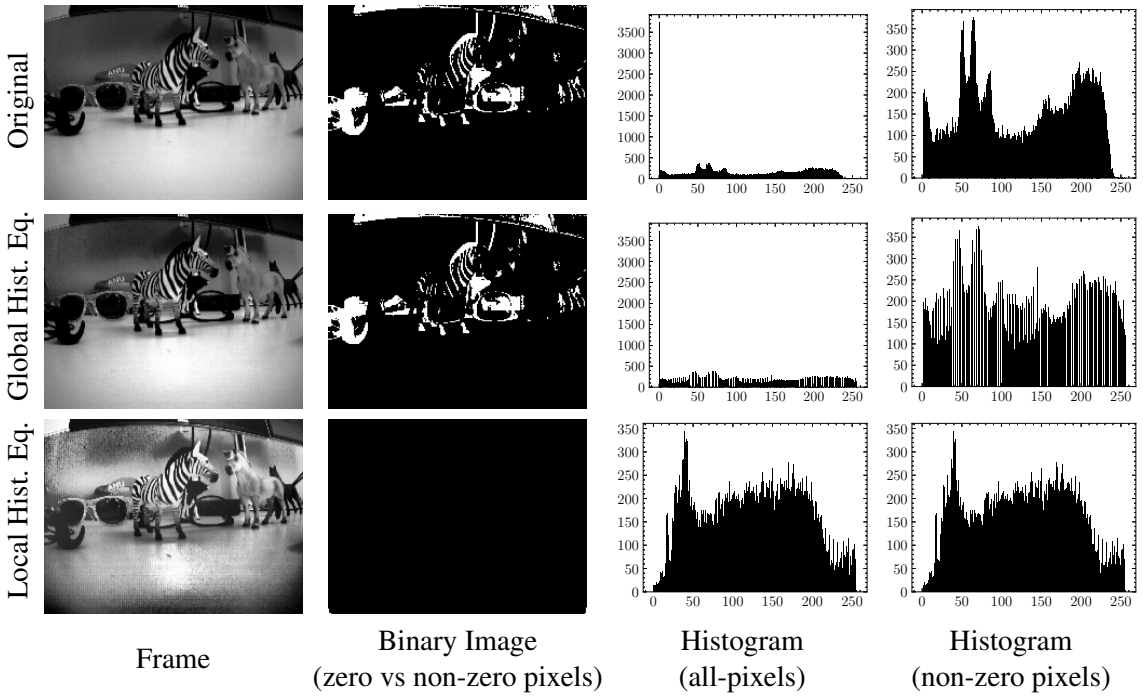


Figure 7.3 An underexposed frame from the HQF dataset, showing the ground truth frame and the corresponding zero intensity areas, together with the effects of global and local histogram equalization.

Conversely, local histogram equalization, as used in E2VID, eliminates zero-intensity regions but produces images with an unrealistic appearance. A further challenge with local histogram equalization is that it has a parameter to be tuned, *i.e.* the local neighborhood for calculation.

It is also worth noting that there is not a consensus in the literature on applying histogram equalization as part of the evaluation procedure. In fact, while some works, including E2VID [12] utilize histogram equalization, much of the recent works tend not to. In [289], this detail is not mentioned at all. In [112], the authors explicitly state not performing any histogram equalization. In [285], the authors even criticize using this step. Given all these reasons, we chose not to employ histogram equalization. We should also emphasize that we include the LPIPS score in our evaluation metrics, which address the challenges posed by intensity variations. LPIPS is renowned for its robustness to such variations, offering a more perceptually accurate assessment than traditional metrics. Its deep learning-based approach, tailored to mimic human visual perception, makes it particularly valuable in scenarios where

intensity inconsistencies might otherwise skew the results of more conventional metrics. This inclusion ensures a more comprehensive and reliable evaluation of image quality, particularly in cases with varying intensity levels.

7.2. Image Quality Results

In this section, we provide the results of our experimental analysis regarding the quality of reconstructed images. This includes qualitative comparisons and quantitative scores with full reference and no reference metrics. Table 7.9 presents the full-reference quantitative results obtained from evaluating the methods on sequences from ECD [280], MVSEC [322], HQF [289], and BS-ERGB [327] datasets. Here, we employ *between-frames* event grouping, match each reconstruction with a ground truth frame, and evaluate reconstruction quality using three evaluation metrics (MSE, SSIM, and LPIPS). We calculate the average values of each metric across all evaluated frames. Our proposed HyperE2VID method achieves state-of-the-art performance in terms of most metrics. On the ECD and MVSEC datasets, it outperforms the second-best method, ET-Net, by a large margin. On the HQF dataset, it delivers results on par with state-of-the-art approaches. On the handheld sequences of the BS-ERGB dataset, it obtains the third-best scores for all three metrics, coming after E2VID+ and ET-Net by small margins. These results demonstrate the effectiveness of the proposed HyperE2VID method, which generates perceptually more pleasing and high-fidelity reconstructions.

In Table 7.10, we present the results of the quantitative analysis on challenging scenarios involving fast camera motion (ECD-Fast [280] and FPVDR [328]), night driving sequences (MVSEC-Night [322]), and high-dynamic range (HDR [12]). Here, we use *fixed-duration* event grouping with a duration of $40\,\mathrm{ms}$ and assess reconstruction quality by using no-reference metrics BRISQUE, NIQE, and MANIQA. Overall, FireNet+ achieves the best results on most of these metrics, with HyperE2VID being the second. FireNet+ excels at HDR sequences, while HyperE2VID obtains good scores on scenarios involving fast camera motion, particularly in the BRISQUE metric. Another method that obtains relatively good

BRISQUE scores is E2VID. The self-supervised method SSL-E2VID obtains the lowest scores on most metrics compared to other methods. Interestingly, ET-Net, the method that achieves the second-best scores on full-reference metrics (*cf.* Table 7.9), performs relatively poorly in these challenging situations. On the other hand, HyperE2VID still performs considerably well across these challenging scenarios, being the best method overall when all full-reference and no-reference metrics are considered.

Table 7.11 presents the quantitative results on our dataset, HUE. We use *fixed-duration* event grouping with a duration of $40\,\mathrm{ms}$ and assess reconstruction quality by using no-reference metrics BRISQUE, NIQE, and MANIQA. The average scores for each of the dataset splits that we present in Section 4.3. (the HUE-City, HUE-Dark, HUE-Day, HUE-Drive, HUE-HDR, and HUE-Indoor) are displayed separately, allowing us to assess the performance of each method for various settings. Even though E2VID obtains the best BRISQUE score in each split, FireNet+ emerges as the better method overall by achieving the best NIQE and MANIQA scores for most of the splits. Our method HyperE2VID obtains the second-best BRISQUE scores for HUE-Dark, HUE-Day, HUE-HDR, and HUE-Indoor, while ET-Net obtains second-best NIQE and MANIQA scores for most of the splits.

In Figures 7.4 to 7.12, we present qualitative results of all aforementioned reconstruction methods, on sample scenes from the ECD [280], MVSEC [322], HQF [289], BS-ERGB [327], ECD-Fast [280], MVSEC-Night [322], HDR [12], and FPVDR [328] datasets. The visual qualities of reconstructions are mostly in line with the quantitative results. Among these eight methods, SPADE-E2VID and SSL-E2VID tend to have the lowest quality, with low-contrast, prominent visual artifacts and blurry regions. Reconstructions of E2VID+ have fewer artifacts, especially at scenes from the HQF dataset (Figures 7.6 and 7.7). E2VID+ also produces nice-looking images for the outdoor scenes of the MVSEC dataset (rows 5 and 6 of Figure 7.5). However, its reconstructions are generally of low contrast and blurry around the edges. ET-Net has better contrast but has more artifacts at textureless regions and around the edges of objects. The reconstructions of HyperE2VID are of high contrast and sharp around the edges. Moreover, the textureless regions are mostly reconstructed with fewer artifacts.

Table 7.9 Full-reference quantitative results on the ECD, MVSEC, HQF, and BS-ERGB datasets. Here we use between-frames event grouping. The best and second best scores are given in **bold** and underlined.

	ECD [280]			MVSEC [322]			HQF [289]			BS-ERGB [327]		
	MSE ↓	SSIM ↑	LPIPS ↓	MSE ↓	SSIM ↑	LPIPS ↓	MSE ↓	SSIM ↑	LPIPS ↓	MSE ↓	SSIM ↑	LPIPS ↓
E2VID [12]	0.179	0.450	0.322	0.225	0.241	0.645	0.099	0.463	0.388	0.139	0.324	0.569
FireNet [91]	0.133	0.459	0.321	0.294	0.198	0.702	0.100	0.422	0.463	0.097	0.330	0.535
E2VID+ [289]	0.070	0.503	0.236	0.132	0.262	0.514	0.036	**0.536**	**0.255**	0.076	**0.374**	**0.433**
FireNet+ [289]	0.062	0.452	0.289	0.219	0.212	0.570	0.045	0.472	0.323	0.091	0.318	0.482
SPADE-E2VID [291]	0.091	0.461	0.337	0.138	0.266	0.591	0.079	0.405	0.514	0.091	0.339	0.643
SSL-E2VID [101]	0.092	0.415	0.380	0.124	0.264	0.694	0.090	0.407	0.496	0.088	0.349	0.628
ET-Net [112]	0.047	0.552	0.224	0.107	0.288	0.489	0.034	0.534	0.268	**0.072**	0.366	0.445
HyperE2VID [334]	**0.033**	**0.576**	**0.212**	**0.076**	**0.315**	**0.477**	**0.032**	0.531	0.261	0.077	0.360	0.446

Table 7.10 No-reference quantitative results on challenging sequences involving fast motion, low light, and high-dynamic range. Here we use fixed-duration event grouping with a duration of 40 ms. The best and second best results are given in **bold** and underlined.

	ECD-Fast [280]			MVSEC-Night [322]			HDR [12]			FPVDR [328]		
	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑
E2VID [12]	14.751	7.004	0.249	**6.106**	6.719	0.310	15.719	4.347	0.310	14.247	6.170	**0.314**
FireNet [91]	19.759	8.203	0.271	23.498	6.319	0.280	21.539	4.085	0.308	21.396	7.609	0.288
E2VID+ [289]	22.239	6.752	0.261	10.737	4.517	0.306	21.340	3.903	0.305	18.677	**4.703**	0.273
FireNet+ [289]	18.118	**5.480**	**0.304**	10.321	**4.470**	0.339	**15.680**	**3.236**	**0.358**	15.508	5.498	0.283
SPADE-E2VID [291]	18.596	9.817	0.284	23.825	7.525	0.309	25.835	5.567	0.286	21.250	8.423	0.310
SSL-E2VID [101]	46.147	9.545	0.201	56.052	12.037	0.212	55.240	6.163	0.228	59.457	11.663	0.187
ET-Net [112]	19.844	7.504	0.283	16.431	5.105	0.324	23.526	3.643	0.335	22.748	5.425	0.284
HyperE2VID [334]	**14.215**	6.764	0.281	8.029	5.294	**0.344**	16.342	3.757	0.345	**14.180**	5.813	0.286

Table 7.11 No-reference quantitative results on six splits of the HUE dataset. Here, we use fixed-duration event grouping with a duration of 40 ms. The best and second best results are given in **bold** and underlined.

	HUE-City			HUE-Dark			HUE-Day		
	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑
E2VID	**4.519**	3.783	0.213	**-0.372**	4.263	0.173	**6.892**	3.680	0.220
FireNet	16.538	**3.399**	0.260	16.826	3.642	0.239	15.945	3.442	0.248
E2VID+	21.125	3.573	0.273	20.657	3.685	0.258	19.572	3.593	0.264
FireNet+	16.126	3.443	**0.368**	14.279	3.687	**0.343**	16.547	**3.302**	**0.351**
SPADE-E2VID	19.191	5.632	0.239	40.584	5.960	0.198	18.060	6.179	0.239
SSL-E2VID+	45.280	5.134	0.211	28.568	7.256	0.243	46.005	5.771	0.195
ET-Net	24.689	3.477	0.313	21.849	**3.503**	0.305	23.378	3.374	0.310
HyperE2VID (ours)	16.858	3.509	0.287	13.890	3.711	0.272	14.781	3.523	0.275

	HUE-Drive			HUE-HDR			HUE-Indoor		
	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑
E2VID	**2.590**	4.313	0.198	**4.578**	3.910	0.204	**1.548**	3.974	0.194
FireNet	13.279	4.080	0.228	13.109	3.506	0.243	14.654	3.700	0.231
E2VID+	16.367	3.913	0.260	17.345	3.477	0.266	13.989	3.748	0.265
FireNet+	9.791	**3.489**	**0.298**	14.916	**3.186**	**0.337**	13.595	**3.303**	**0.317**
SPADE-E2VID	22.143	6.133	0.253	20.207	7.232	0.251	31.330	6.942	0.220
SSL-E2VID+	65.749	7.886	0.173	48.331	5.922	0.187	47.613	7.384	0.212
ET-Net	25.097	3.770	0.283	19.274	3.222	0.311	16.807	3.498	0.298
HyperE2VID (ours)	18.804	4.381	0.251	11.931	3.510	0.271	12.343	3.918	0.262

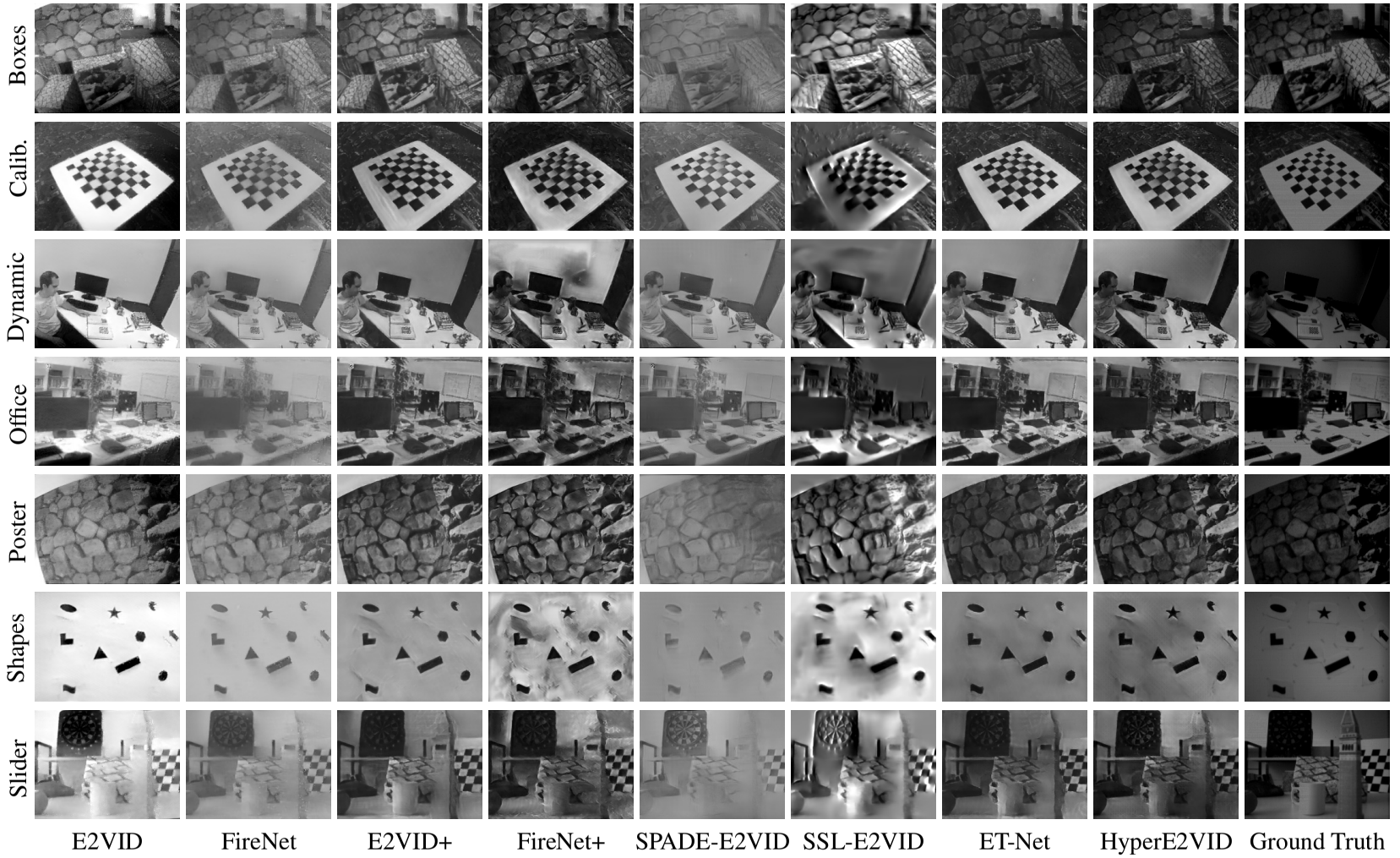


Figure 7.4 Qualitative comparisons on sequences from ECD. The sequences presented, from top to bottom, are `boxes_6dof`, `calibration`, `dynamic_6dof`, `office_zigzag`, `poster_6dof`, `shapes_6dof`, and `slider_depth`.

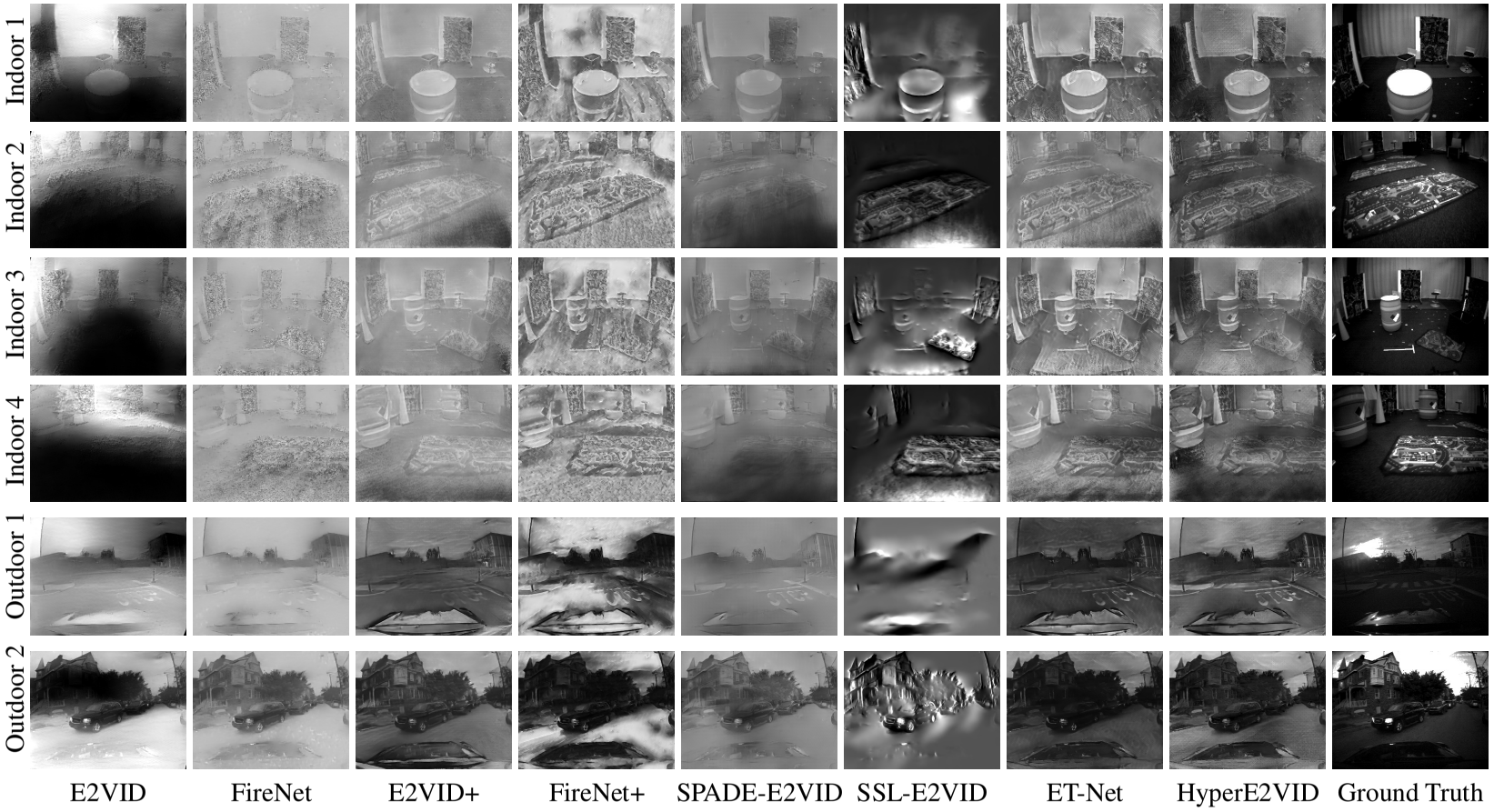


Figure 7.5 Qualitative comparisons on sequences from MVSEC. The sequences presented, from top to bottom, are `indoor_flying1_data`, `indoor_flying2_data`, `indoor_flying3_data`, `indoor_flying4_data`, `outdoor_day1_data`, and `outdoor_day2_data`.

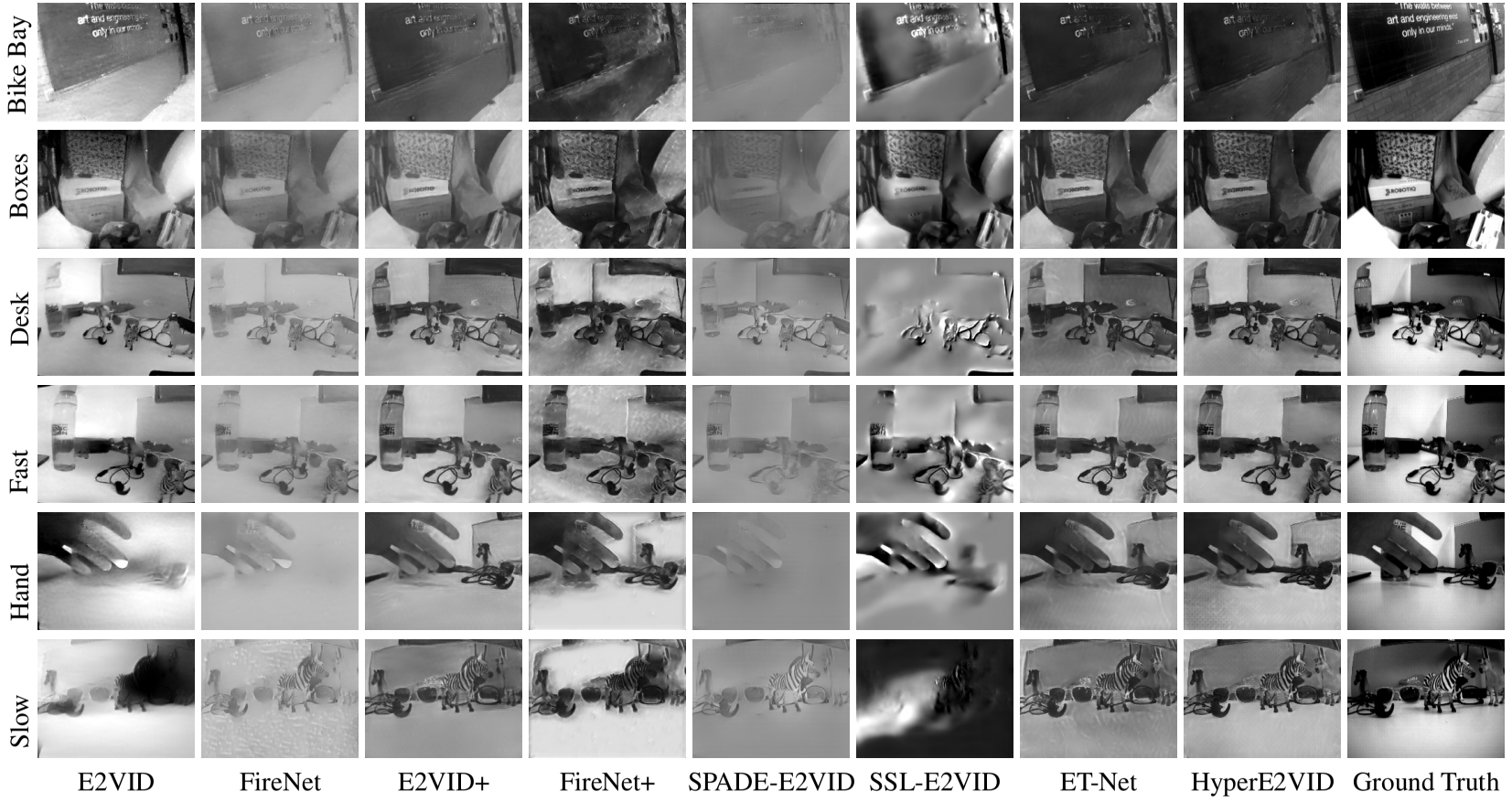


Figure 7.6 Qualitative comparisons on sequences from HQF. The sequences presented, from top to bottom, are `bike_bay_hdr`, `boxes`, `desk`, `desk_fast`, `desk_hand_only`, and `desk_slow`.

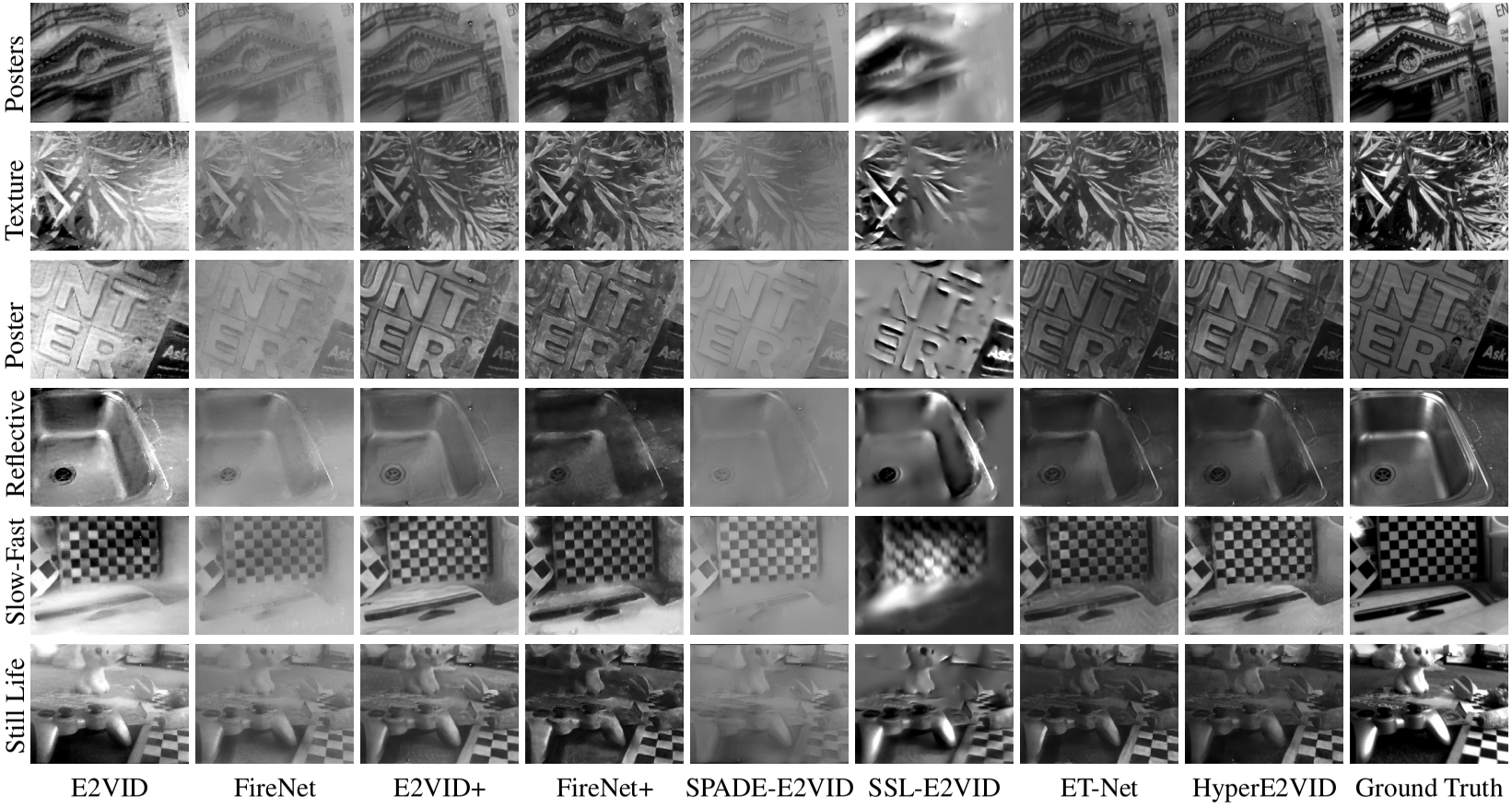


Figure 7.7 Qualitative comparisons on sequences from HQF. The sequences presented, from top to bottom, are `engineering_posters`, `high_texture_plants`, `poster_pillar_1`, `reflective_materials`, `slow_and_fast_desk`, and `still_life`.

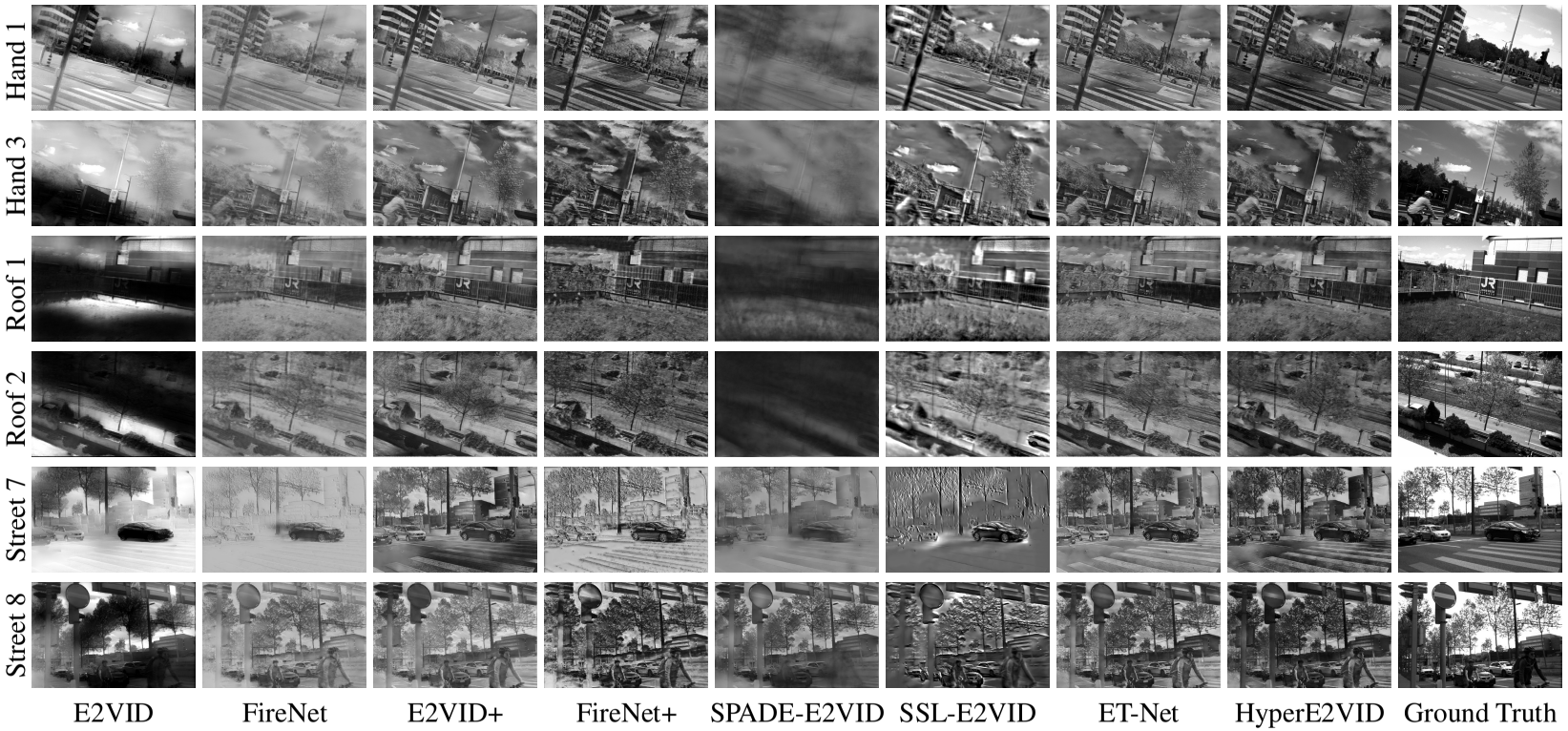


Figure 7.8 Qualitative comparisons on sequences from BS-ERGB. The sequences presented, from top to bottom, are `may29_handheld_01`, `may29_handheld_03`, `may29_rooftop_handheld_01`, `may29_rooftop_handheld_02`, `street_crossing_07`, and `street_crossing_08`.

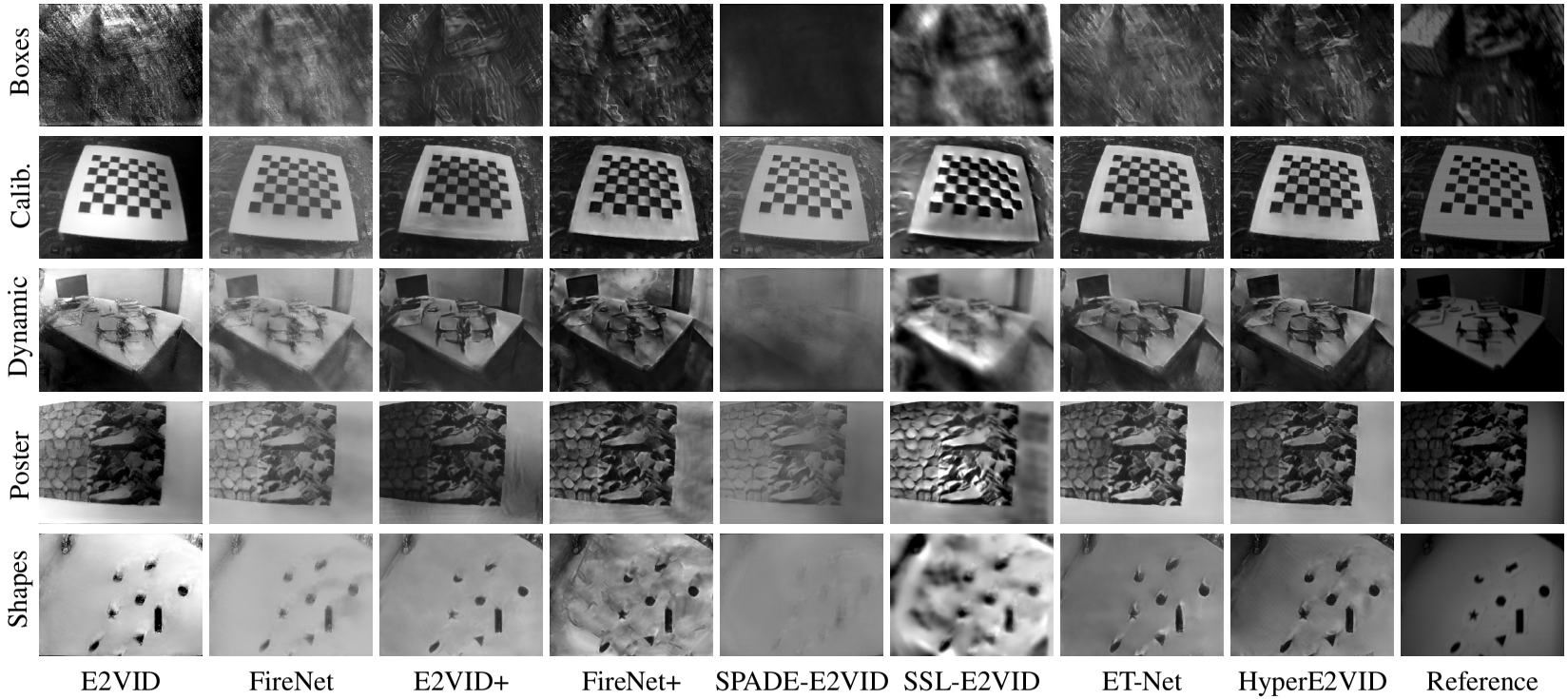


Figure 7.9 Qualitative comparisons on the fast parts of the ECD. The sequences presented, from top to bottom, are `boxes_6dof`, `calibration`, `dynamic_6dof`, `poster_6dof`, and `shapes_6dof`.

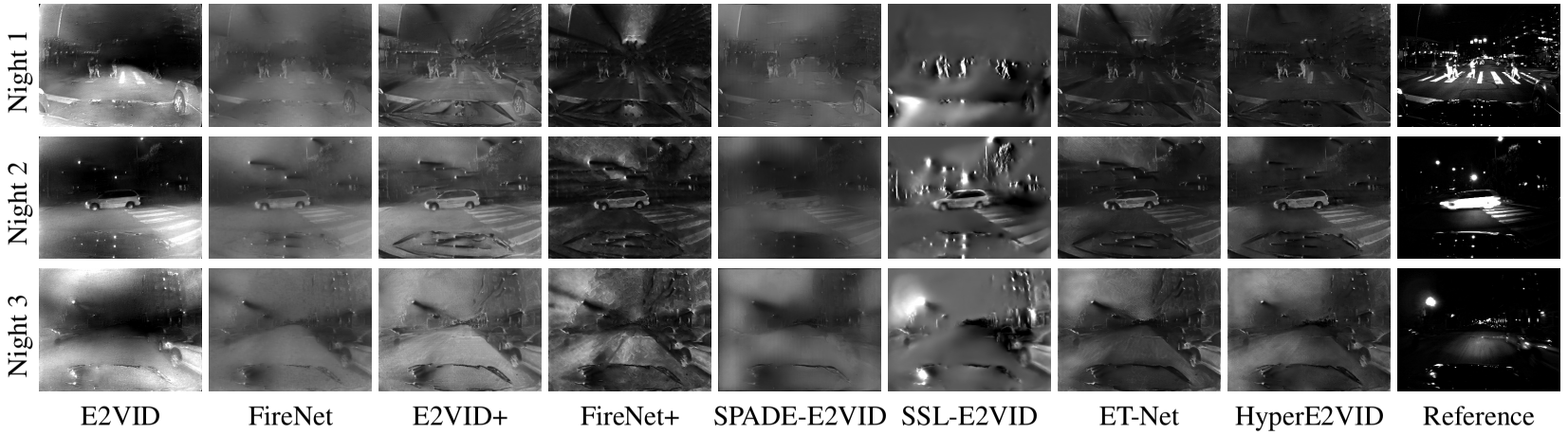


Figure 7.10 Qualitative comparisons on the night sequences of MVSEC. The sequences presented, from top to bottom, are `outdoor_night1_data`, `outdoor_night2_data`, and `outdoor_night3_data`.

Figure 7.11 Qualitative comparisons on `selfie` sequence from HDR dataset[12].

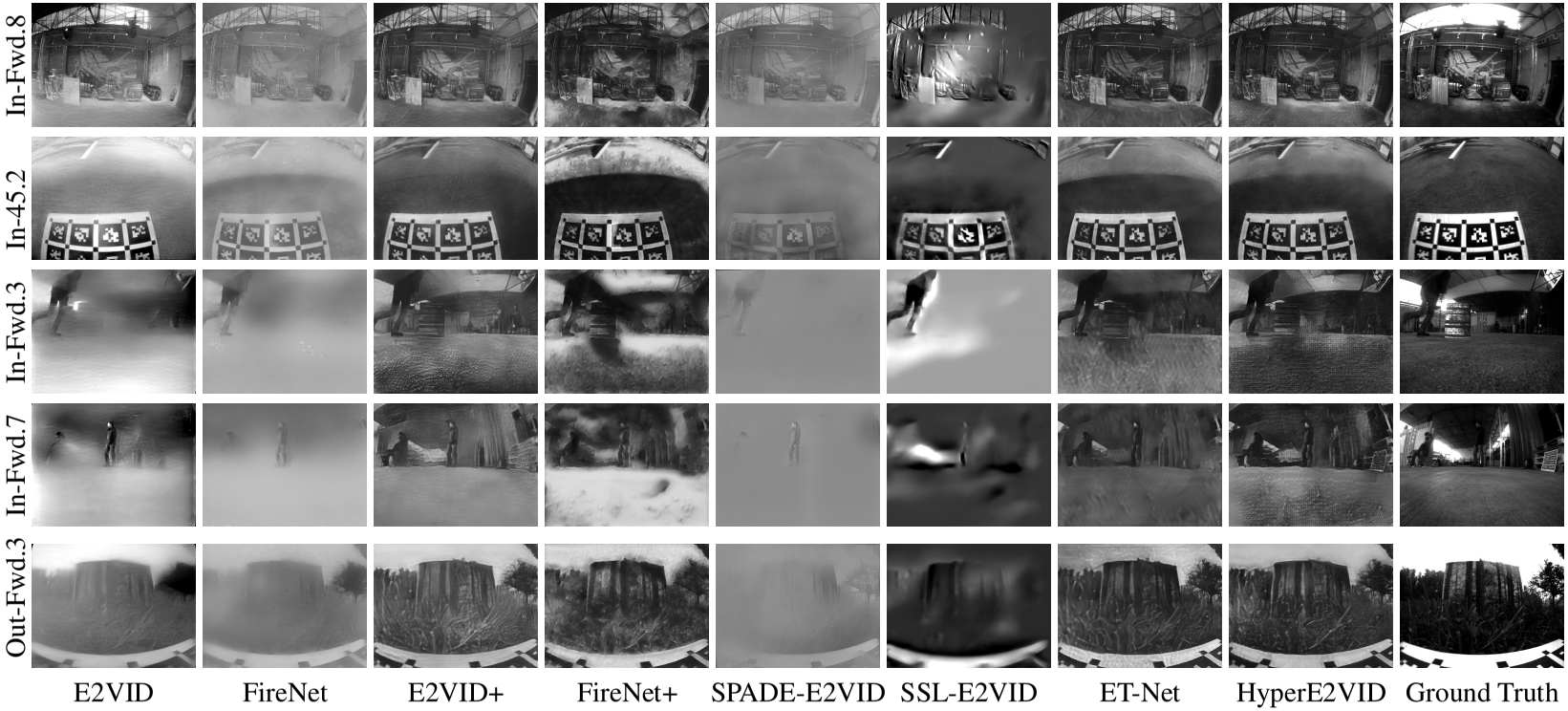


Figure 7.12 Qualitative comparisons on sequences from FPVDR. The sequences presented, from top to bottom, are `indoor_forward_8_davis`, `indoor_45_2_davis_with_gt`, `indoor_forward_3_davis_with_gt`, `indoor_forward_7_davis_with_gt`, and `outdoor_forward_3_davis_with_gt`.

Figures 7.13 to 7.19 present the qualitative results on the HUE-City, HUE-Dark, HUE-Day, HUE-Drive, HUE-HDR, and HUE-Indoor splits of our HUE dataset. By visually assessing the performance of each method across these splits, it's evident that each technique exhibits unique strengths and weaknesses.

The HUE-City results presented in Figure 7.13 reveal significant differences among the methods. E2VID stands out for its sharp edges and high contrast, although it struggles with graininess in textureless regions. FireNet+ offers the highest contrast, yet it introduces unrealistic artifacts. Reconstructions of SPADE-E2VID are very blurry most of the time, although it performs well for the well-lit and slow-motion sequence presented in the first row. SSL-E2VID generates reconstructions with high contrast, but they also have unrealistic color variations and blurry edges. HyperE2VID performs well in terms of sharpness and realism but occasionally suffers from checkerboard artifacts.

For scenarios with challenging lighting conditions such as the HUE-Dark and HUE-HDR splits, the methods respond differently. By looking at the results of HUE-Dark in Figure 7.15, we can see that most methods, such as E2VID and FireNet, struggle with edge definition and contrast. FireNet+ offers high contrasts, but the reconstructions are mostly unnatural, which is most obvious in the face sequence presented in the seventh row of Figure 7.15. HyperE2VID maintains better edge sharpness, despite presenting occasional checkerboard patterns (*e.g.* road surface in the sixth row). ET-Net presents more artifacts in low-light scenarios, as can be seen in examples like the lake surface in the fourth row of the same figure. The high-dynamic range scenarios prove particularly challenging for E2VID: the reconstruction of it in the last row of Figure 7.14 shows no details in darker regions. HyperE2VID emerges as the best method for the challenging cases of HUE-HDR, by presenting reconstructions with high contrast, sharp edges, and minimal artifacts.

Interestingly, E2VID and FireNet+ have issues in well-lit sequences of HUE-Day as well, despite getting the best quantitative scores. The reconstructions of E2VID presented in the first column of Figure 7.16 often have unrealistic dark regions and blurry edges. The reconstructions of FireNet+, which can be seen in the fourth column of the same figure,

also suffer from unrealistic intensity variations and overly textured regions. As a striking example, most methods fail to reconstruct tree leaves in the courtyard sequence, presented in the third row of Figure 7.16. Only ET-Net and HyperE2VID offer good reconstructions for this case, while HyperE2VID is more realistic and has higher contrast.

In more dynamic and complex scenes, such as those in the HUE-Drive and HUE-Indoor, the limitations of some methods become more pronounced. FireNet+, for instance, is severely affected by light trails in the night driving sequences presented in Figure 7.17, while other methods like FireNet, SPADE-E2VID, and SSL-E2VID consistently struggle with lower-quality reconstructions. ET-Net and HyperE2VID generally provide better results in Figures 7.17 to 7.19. HyperE2VID shows fewer artifacts and is more realistic in general, but it also has some problematic cases, such as the unnatural intensity variation in the bookshelf sequence presented in the first row of Figure 7.18.

The qualitative assessment highlights a consistent challenge across methods in balancing contrast, edge sharpness, and realistic intensity representation, suggesting that further refinement is needed to enhance overall image quality. Another implication of these observations is that no-reference quantitative metrics do not seem to be sufficient to cover different types of degradation seen in these examples.

Figure 7.13 Qualitative comparisons on HUE-City sequences `city_day_1`, `city_day_3`, `city_day_4`, `city_day_5`, and `city_twilight_1`.

Figure 7.14 Qualitative comparisons on HUE-HDR sequences `hdr_plants` and `hdr_terrace_sun_2`.

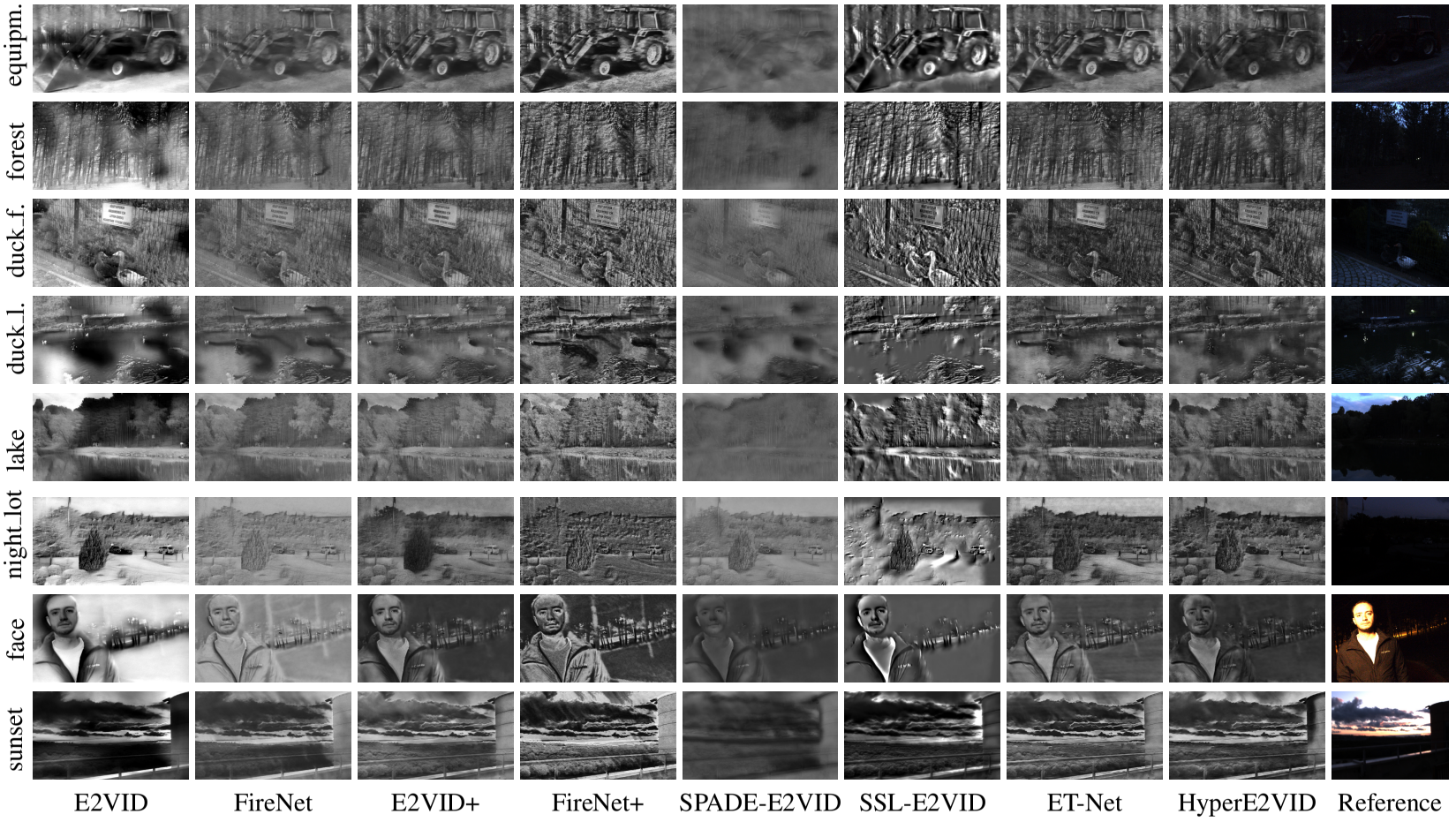


Figure 7.15 Qualitative comparisons on HUE-Dark sequences `dark_equipment`, `dark_forest_1`, `duck_fence`, `duck_lake_4`, `lake_4`, `night_parking_lot`, `person_face`, and `terrace_sunset`.

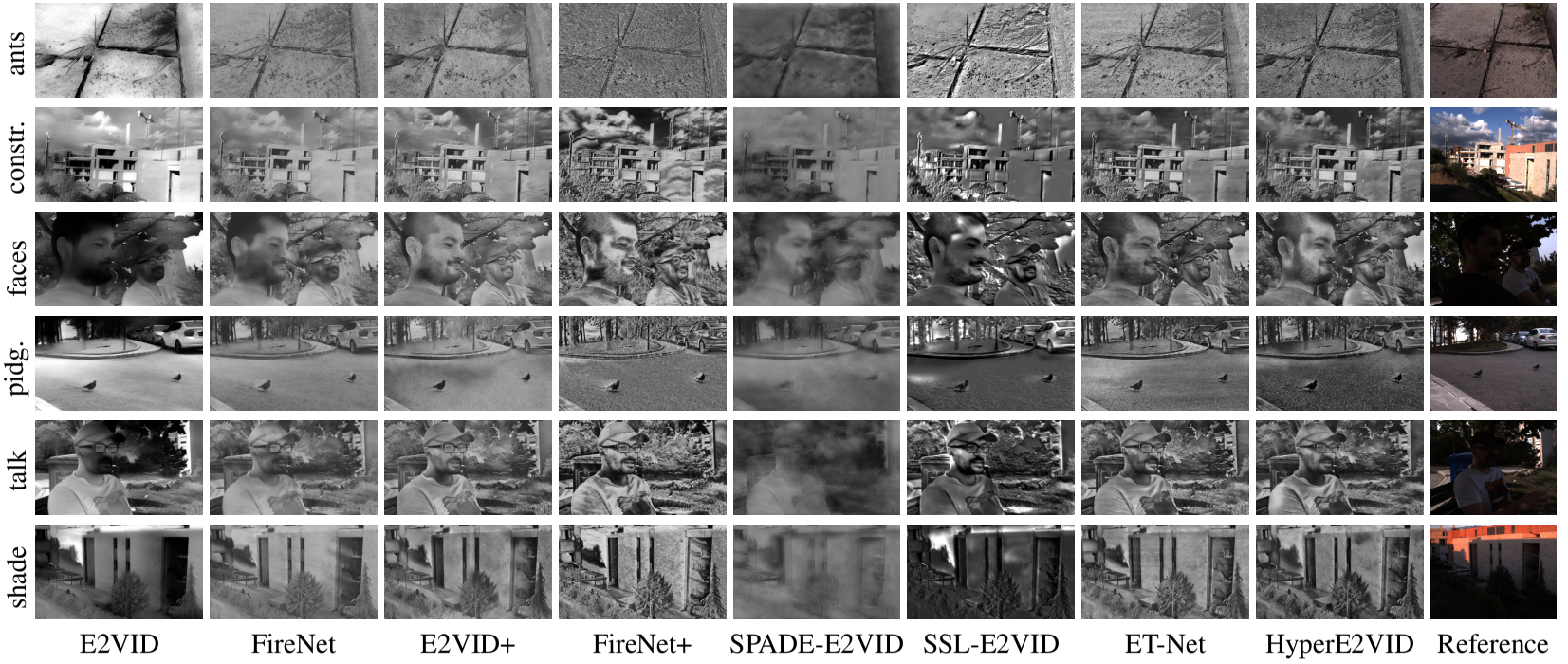


Figure 7.16 Qualitative comparisons on HUE-Day sequences `ants`, `construction`, `day_close_faces`, `pidgeons_close`, `day_dynamic_talk`, and `sun_shade_building`.

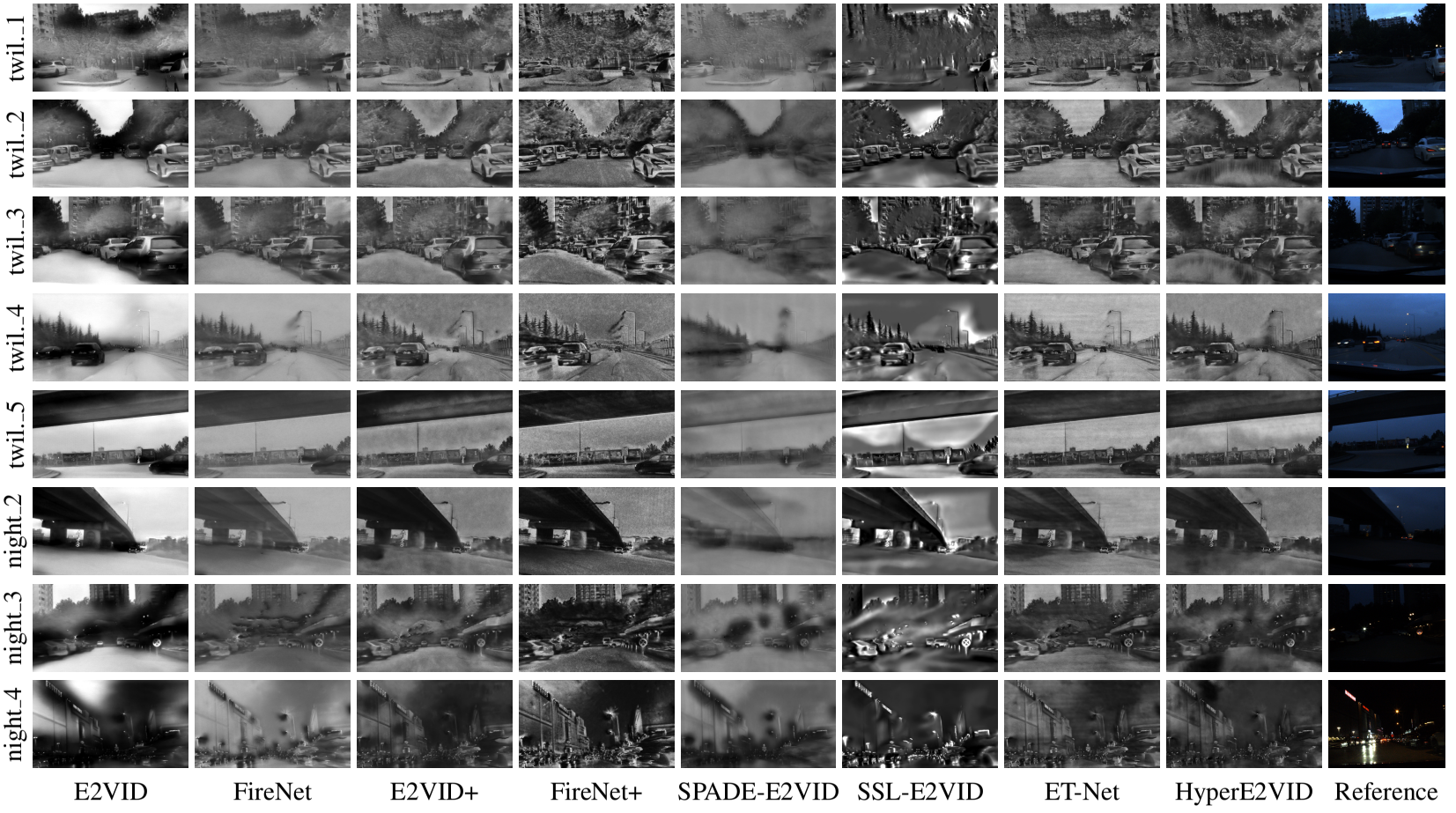


Figure 7.17 Qualitative comparisons on HUE-Drive sequences `drive_twilight_1`, `drive_twilight_2`, `drive_twilight_3`, `drive_twilight_4`, `drive_twilight_5`, `drive_night_2`, `drive_night_3`, and `drive_night_4`.

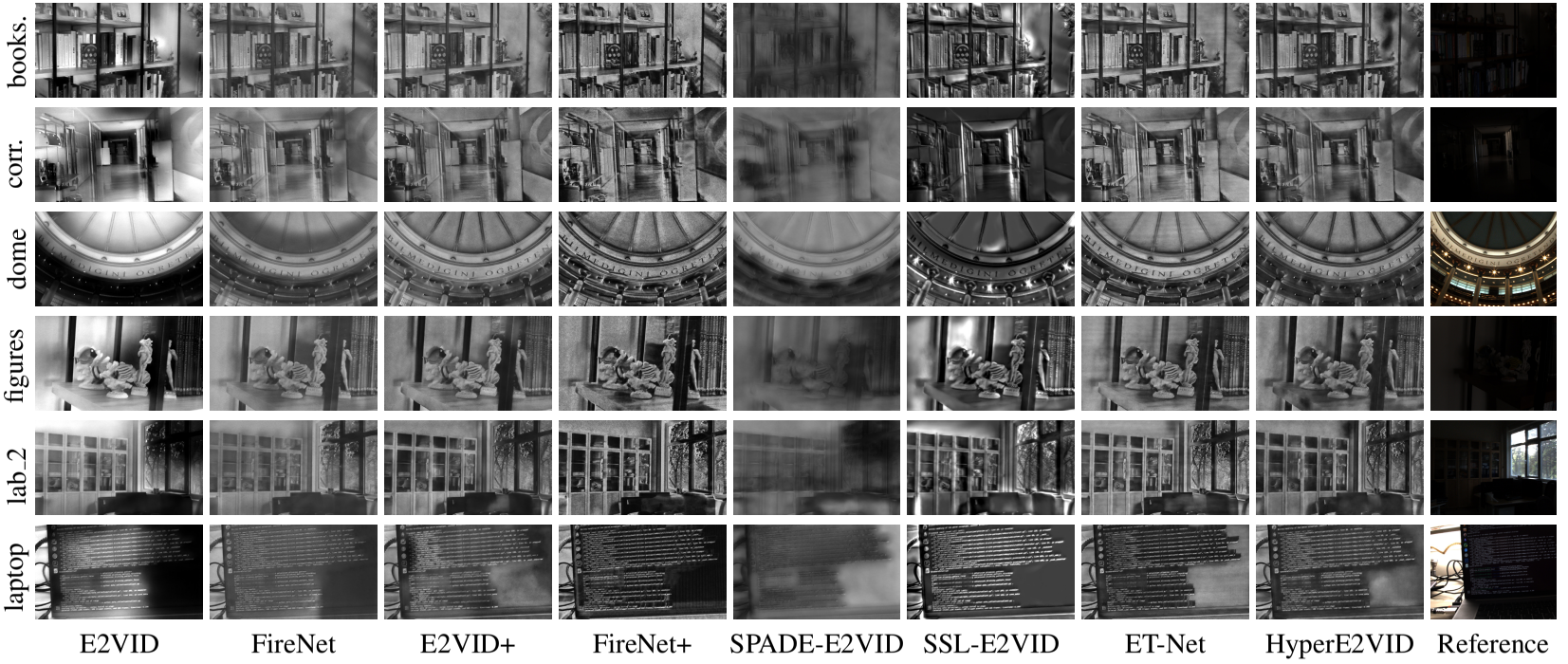


Figure 7.18 Qualitative comparisons on HUE-Indoor sequences `bookshelves`, `corridor`, `dome`, `figures_classics`, `lab_2`, and `laptop`.

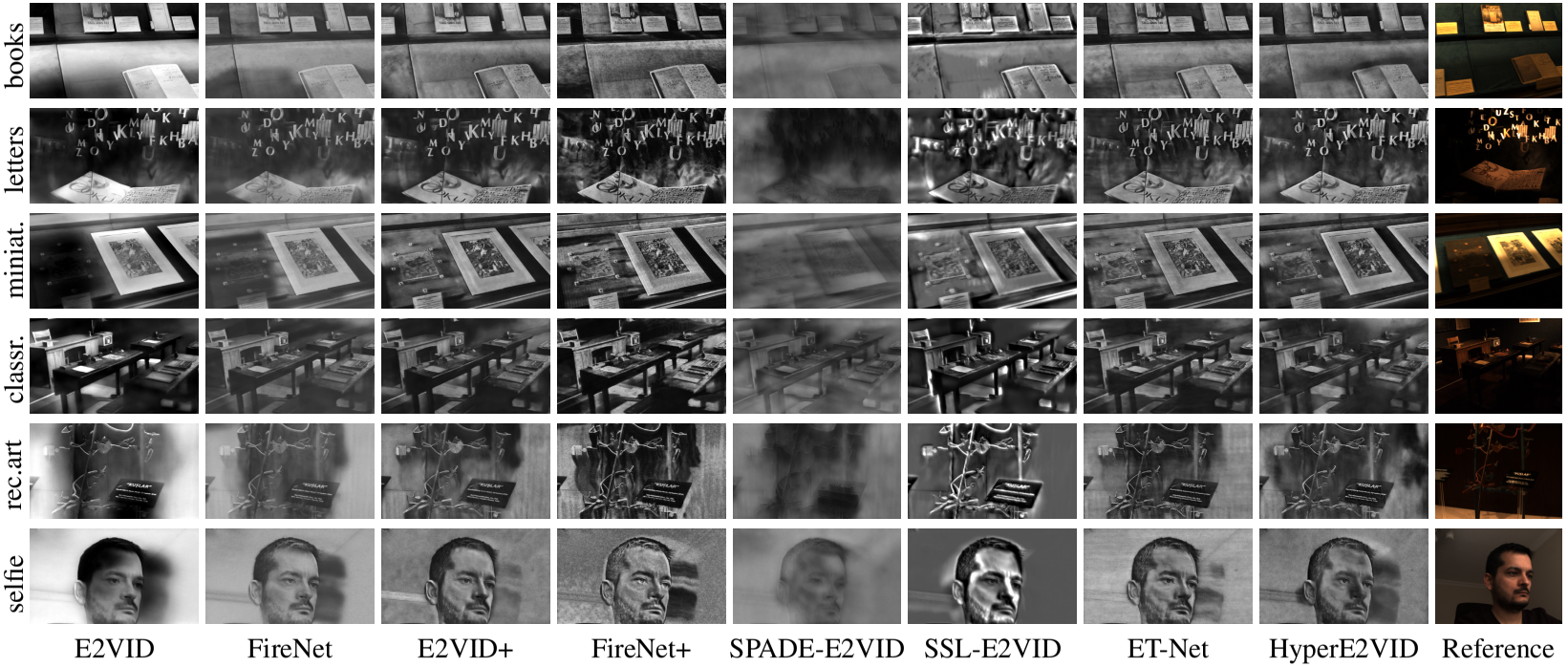


Figure 7.19 Additional qualitative comparisons on HUE-Indoor sequences `old_books`, `letters`, `miniature`, `old_classroom_2`, `recycle_art`, and `selfie`.

7.3. Color Reconstructions

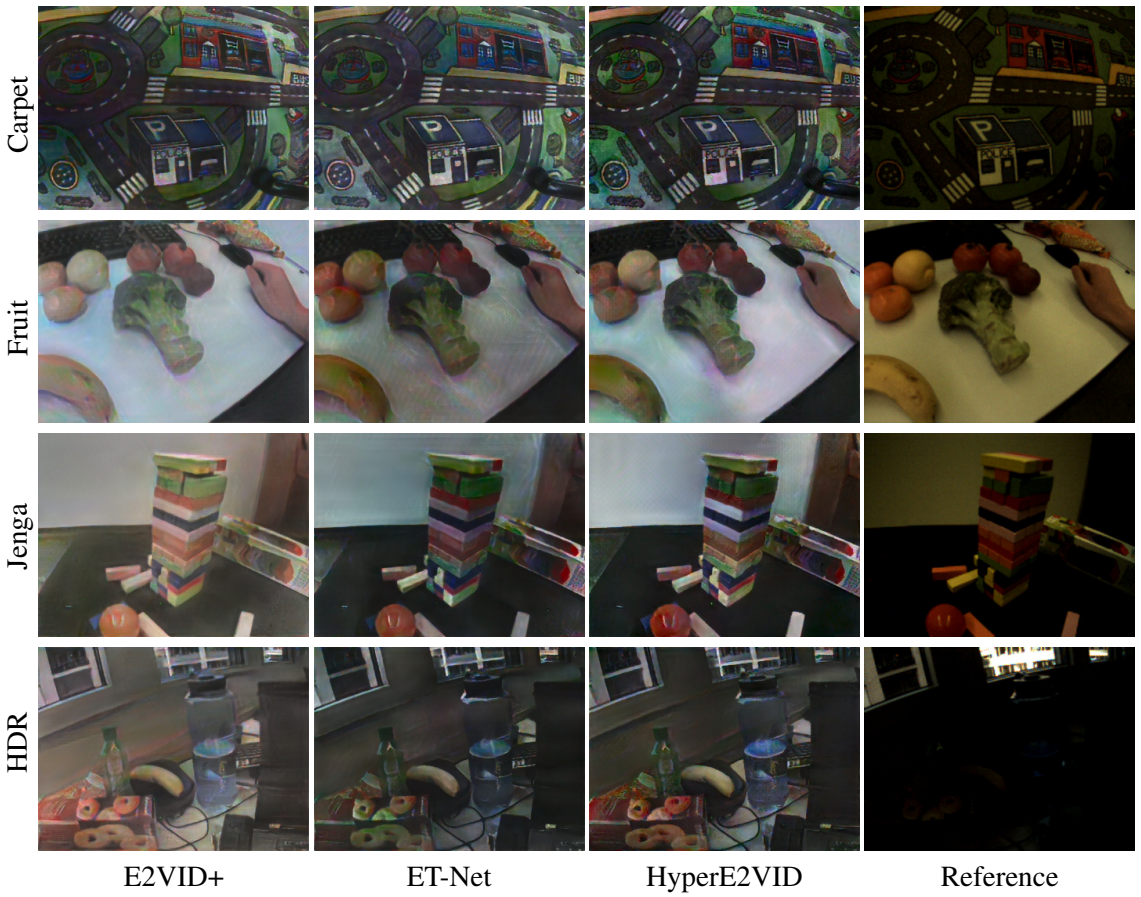


Figure 7.20 Color image reconstructions on CED. HyperE2VID excels in reconstructing visually appealing scenes from the CED dataset, including those with colorful objects and HDR scenarios, outperforming E2VID+ and ET-Net in visual quality.

In Figure 7.20, we show color reconstructions from HyperE2VID alongside those from two top-performing competitors, E2VID+ and ET-Net, using sample scenes from CED. The results demonstrate HyperE2VID's ability to produce color images of superior quality. These images exhibit sharp edges, minimal artifacts, and authentic colors, even in challenging lighting conditions, such as the high-dynamic-range (HDR) scene displayed in the last row.

7.4. Robustness Analysis

Reconstructing images from events poses a challenge due to its intricate nature, which is influenced by numerous variables impacting method performances. A technique excelling

under certain conditions might not be universally applicable if these variables are subject to change. Therefore, it is essential to assess method sensitivity to these variables and verify their performance across varying conditions. In our robustness analysis, we investigate the impact of four critical ones: *image reconstruction rate*, *event tensor sparsity*, *temporal irregularity*, and *event rate*. We employ commonly used sequences from the ECD, MVSEC, and HQF datasets as mentioned in Section 7.1.1. and utilize the LPIPS metric to evaluate the results. We provide detailed descriptions of these experiments below:

Reconstruction rate. To evaluate the impact of changing frame reconstruction rates on each method's performance, we conduct experiments using fixed-duration grouping, which generates a fixed number of frames per second. We perform ten experiment runs, each with a different event grouping duration ranging from 10 ms to 100 ms, corresponding to frame reconstruction rates from 10 FPS to 100 FPS. We use a tolerance of 1 ms to match the reconstructions with ground truth frames. We then compute the average LPIPS values for each experiment run and method to determine their performance under different frame rates.

Tensor sparsity. To analyze how the sparsity of event tensors affects the performance of each method, we carry out experiments utilizing fixed-number grouping and a tolerance of 1 ms to match the reconstructions with ground truth frames. With this grouping approach, each group contains the same number of events, resulting in event tensors with the same sparsity level. Specifically, we conduct nine different experiment runs, with event numbers ranging from 5K to 45K. We then compute the mean LPIPS scores for each experiment run and method.

Note that when the scene contains slow motion or little texture, the event rate would be lower, and using fixed-number grouping would result in event groups that span a large temporal window. Furthermore, the motion or texture captured by the event camera might be contained in a small region of pixels rather than being homogeneously distributed to all of the sensor area. In that case, the temporal discretization performed in the event representation (to a fixed number of temporal bins) means more compression of the temporal information, and

this might result in reconstruction artifacts such as saturation or blur in these regions. The tensor sparsity experiments aid us in assessing each method's robustness to these situations.

Temporal irregularity. To evaluate the robustness of each method in generating frames at irregular intervals, we conduct experiments by removing a certain percentage of ground truth frames from each sequence and by using *between-frames* event grouping to group events between the remaining frames. In particular, we conduct ten experiment runs with different discarding ratios ranging from 0.0 (standard case) to 0.9. We then calculate the mean LPIPS scores obtained for each experiment run for each method.

Event rate. To evaluate the robustness of the methods to varying event rates, we employ *between-frames* event grouping and collect statistics on event rates, measured in events per second, for each group. We then reconstruct intensity images using each method based on the event groups and calculate LPIPS scores for each time step. We divide the event rate spectrum into ten equally spaced bins and compute the mean LPIPS scores for each bin and method. This enables us to assess the performance of each method under different event rate conditions and determine which methods are most robust to changes in event rate.

For all these experiments, we employ a tolerance of 1 ms to match the reconstructions with ground truth frames and calculate LPIPS scores whenever there is a match. We then compute mean LPIPS scores for each method and experiment. Figure 7.21 shows plots of mean LPIPS scores for these four experiments: robustness to image reconstruction rate, event tensor sparsity, temporal irregularity, and event rate. In these plots, we omit methods with lower image quality scores for clarity and focus on the four best-performing methods: E2VID+, FireNet+, ET-Net, and HyperE2VID. The results demonstrate the superiority of the proposed HyperE2VID architecture for generating high-quality reconstructions over a wide range of settings.

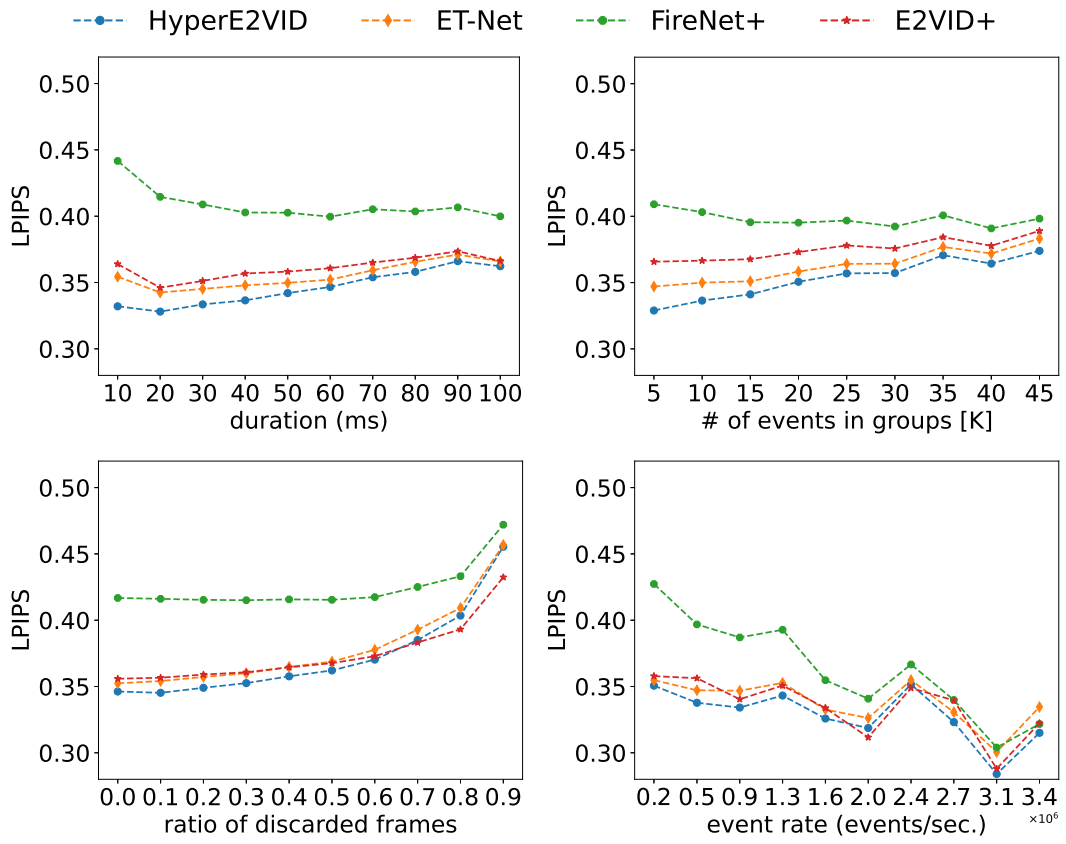


Figure 7.21 Robustness analysis. We investigate how factors including image reconstruction rate (top-left), event tensor sparsity (top-right), temporal irregularity (bottom-left), and event rate (bottom-right) affect the performance of the event-based video reconstruction methods.

7.5. Analysis on Two Other Challenging Scenarios

Our proposed framework EVREAL and the experimental setup described in Section 7.1., with the utilization of no-reference metrics in particular, helped us to quantitatively assess the qualities of reconstructed images on challenging sequences involving fast motion, low light, and high-dynamic range. In this section, we further evaluate the quality of reconstructions in two other challenging scenarios: high frame rate video generation (200 to 5000 FPS) and reconstruction during motionless periods. Details of these experiments are presented below:

High Frame Rate Video Reconstruction. For high frame rate video reconstruction, Rebecq *et al.* [12] suggested a method that groups a fixed number of events and runs

multiple reconstructions in parallel, each with a slight temporal shift. This technique, however, necessitates the selection of an event count and a temporal shift value. Also, it involves conducting numerous separate reconstructions to produce a set of videos, which are then merged by reordering frames and subjected to temporal filtering to mitigate flickering, ultimately yielding a video with a variable frame rate. In contrast, we employ a simple approach with fixed-duration event grouping for generating videos with high FPS, without the need for temporal shifts or parallel reconstructions, facilitating the generation of a high and constant frame rate video. The temporal window for event grouping is straightforwardly determined based on the desired frame rate, using the formula 1/FPS, where a smaller window correlates with a higher FPS.

In Figure 7.22, we present frames corresponding to the first second of the `slider_depth` sequence from the ECD dataset, taken from videos reconstructed at $200\,\mathrm{Hz}$, $500\,\mathrm{Hz}$, $1\,\mathrm{kHz}$, $2\,\mathrm{kHz}$, and $5\,\mathrm{kHz}$, which are generated by using temporal windows of $5\,\mathrm{ms}$, $2\,\mathrm{ms}$, $1\,\mathrm{ms}$, $500\,\mu\mathrm{s}$, and $200\,\mu\mathrm{s}$, respectively. The results reveal that most event-based video reconstruction networks from existing literature begin to falter in visual quality when the FPS exceeds one thousand, as the event voxel grid statistics start to diverge from the conditions these methods trained under. HyperE2VID, however, consistently produces high-contrast, sharp reconstructions, even at frame rates of several thousand frames per second. Owing to its dynamic network architecture, HyperE2VID adeptly adjusts to the varying event statistics, thus maintaining superior visual quality in high FPS video output.

Reconstruction During Still Periods. Another challenging case for event-based video reconstruction is the stationary sections in event sequences since the event rate drastically reduces, with only noise events being generated by the camera. Here, we qualitatively analyze the reconstruction quality of HyperE2VID and other methods during these motionless periods by presenting their reconstructions in Figure 7.23. We consider a segment from the UZH-FPV Drone Racing dataset, where the drone lands on a board with ArUco markers and stops. For each method, we present reconstructions from the initial time just after the drone stops in the leftmost column and three more reconstructions at one-second

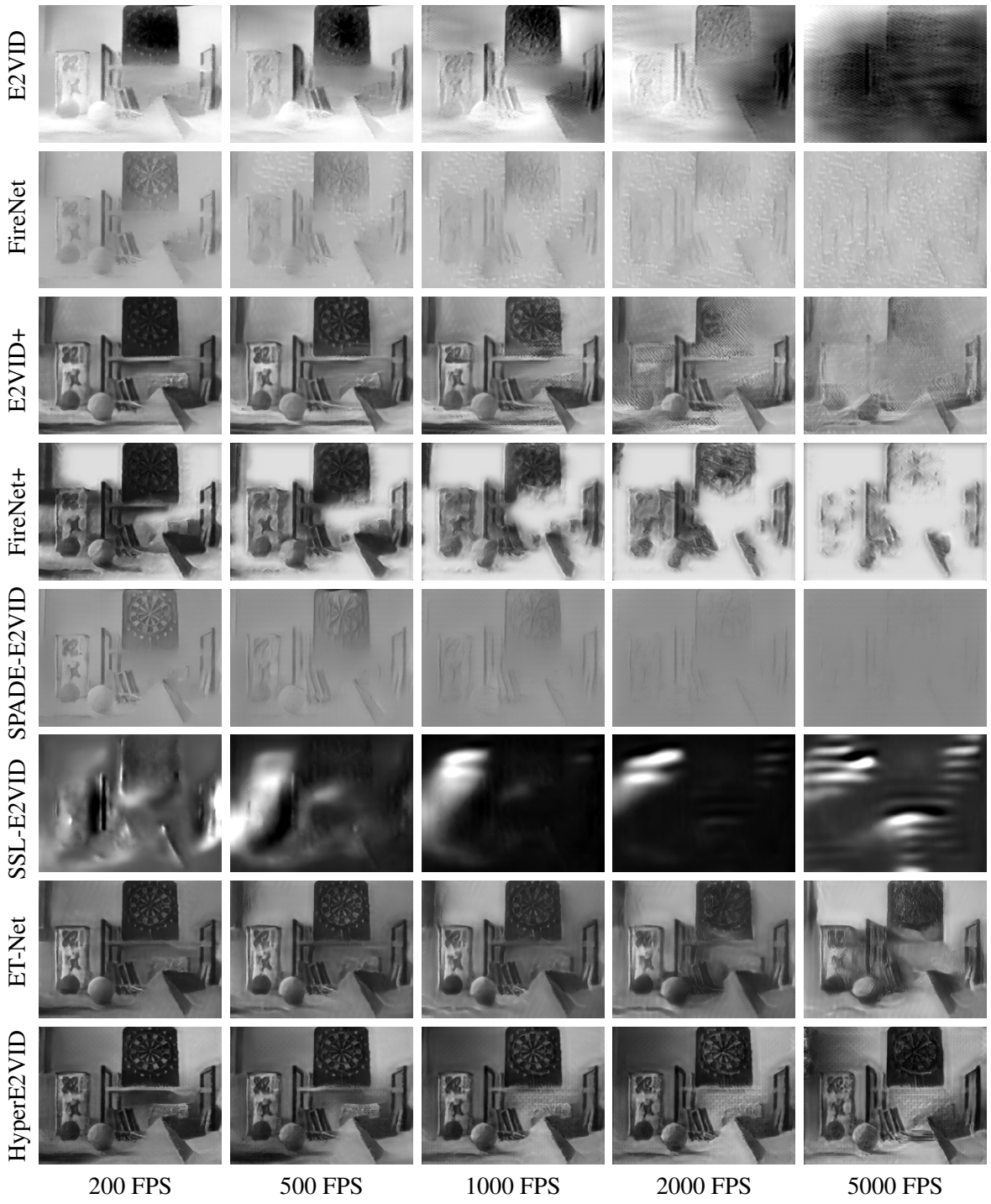


Figure 7.22 High frame rate video synthesis. Here we present frames corresponding to the first second of the `slider_depth` sequence from the ECD dataset, taken from videos reconstructed at 200 Hz, 500 Hz, 1 kHz, 2 kHz, and 5 kHz, which are generated by using temporal windows of 5 ms, 2 ms, 1 ms, 500 µs, and 200 µs, respectively.

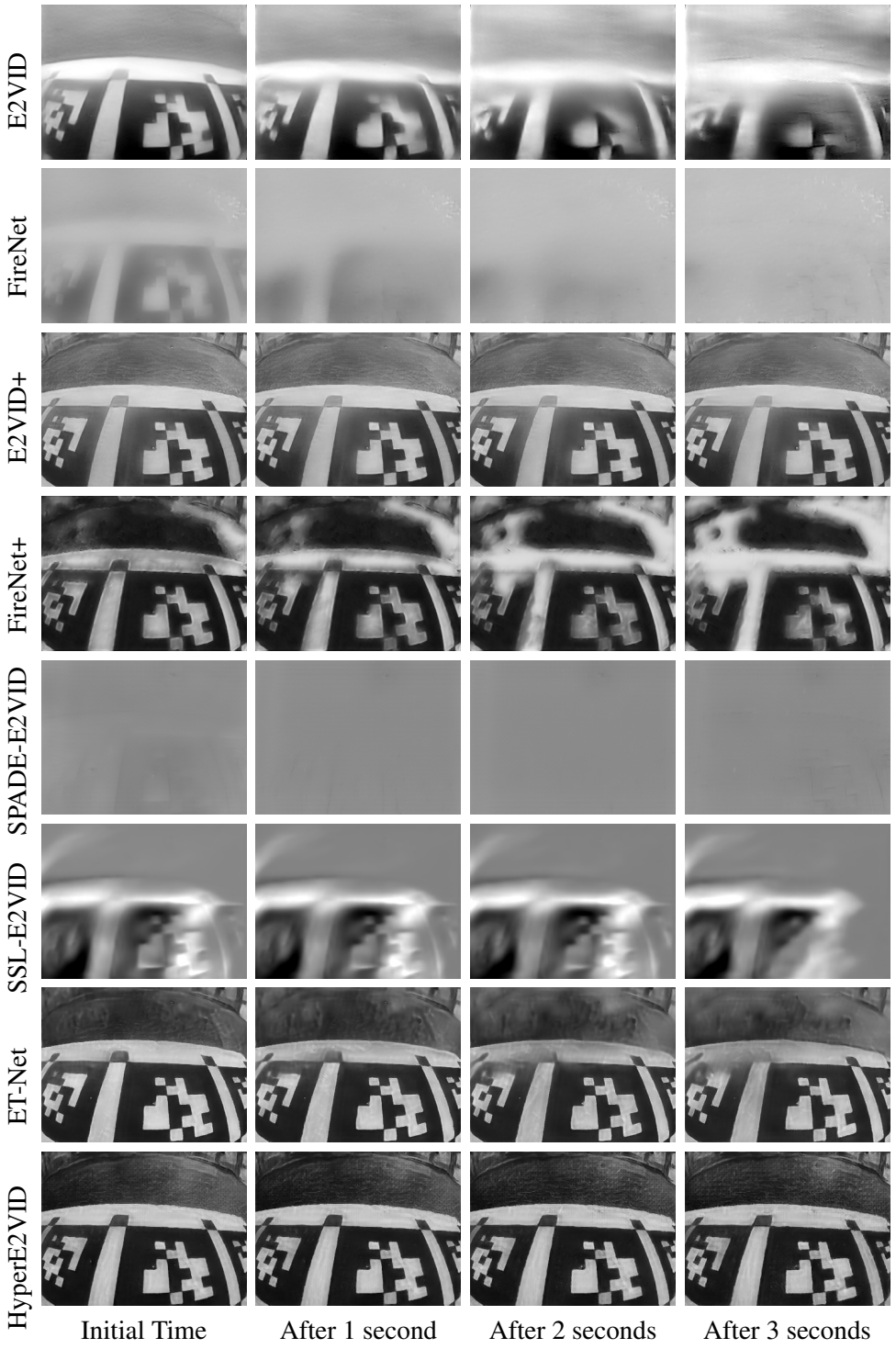


Figure 7.23 Assessing reconstruction quality in motionless sections. Here, we consider a segment from the UZH-FPV Drone Racing dataset, where the drone lands on a board with ArUco markers and stops. For each method, we present reconstructions just after the drone stops in the leftmost column and three more reconstructions at one-second intervals in subsequent columns.

intervals in subsequent columns. The desired functionality for methods is to retain their most recent reconstructions during the pause segment, but most of them start to generate intensity images with degraded quality within a few seconds by gradually decaying images and revealing artifacts such as blurry and bleeding edges. On the other hand, the results presented in the last row of Figure 7.23 demonstrate HyperE2VID's ability to preserve its high contrast and sharp reconstructions during the motionless segments, thanks to its dynamic network architecture, allowing it to adapt to highly varying event data.

7.6. Analysis on Downstream Tasks

One of the primary motivations behind reconstructing images from events is that these reconstructed images can be used for downstream tasks, utilizing frame-based computer vision methods. As explained in Section 5.8., with EVREAL, we perform analysis on three such downstream tasks: object detection, image classification, and camera calibration. Table 7.12 shows the quantitative results of image reconstruction methods on these three downstream tasks, including results using ground truth intensity frames as a baseline for comparison. For the image classification task, since the N-Caltech101 dataset does not include intensity images, we leave the accuracy field blank in the last row. The evaluation metrics employed are AP (Average Precision) for object detection, accuracy for image classification, and MAPE (Mean Absolute Percentage Error) for camera calibration.

E2VID achieves the highest score on image classification and the second highest score on object detection, with FireNet being the third and second best for these tasks, respectively. E2VID+ obtains a lower score on object detection than these two methods, but still performs well on image classification, achieving the second best score. However, its performance on camera calibration is significantly worse than the first two methods. Conversely, SPADE-E2VID has substantially lower performance on image classification while performing decently well on camera calibration. Even though ET-Net obtains good scores in full-reference image quality metrics (*c.f.* Table 7.9), its downstream task performance is relatively low compared to other methods. HyperE2VID performs well on

Table 7.12 Quantitative results on downstream tasks. The best and second best results are highlighted in **bold** and underlined.

	Object Detection	Image Classification	Camera Calibration
Methods	AP (%)	Accuracy (%)	MAPE (%)
E2VID [12]	53.67	**75.99**	3.26
FireNet [91]	**64.11**	67.93	2.57
E2VID+ [289]	52.15	70.47	6.26
FireNet+ [289]	28.83	47.17	**1.70**
SPADE-E2VID [291]	35.80	19.53	2.89
SSL-E2VID [101]	52.03	60.68	8.58
ET-Net [112]	47.87	66.52	3.88
HyperE2VID [334]	40.86	66.26	1.92
Ground Truth Frames	72.36	–	4.53

camera calibration, achieving the second best score. However, its performance on image classification and object detection is relatively low.

In summary, the results show that choosing a method may depend on the specific downstream task: FireNet is superior for night-time vehicle detection, E2VID is the best method for image classification, and FireNet+ is the best performer for camera calibration. The object detector achieves the highest score when run on the original intensity images, meaning that intensity images provide a strong baseline for the object detection task, and further research is needed to improve object detection performance on reconstructed frames. On the other hand, using intensity sequence does not give the best scores on the camera calibration task, demonstrating the potential of reconstructing intensity images from events for downstream tasks.

7.7. Computational Complexity

We also analyze the computational complexity of each method by considering three metrics: (1) the number of model parameters, (2) the number of floating point operations (FLOPs), and (3) inference time. The number of parameters is an important metric that indicates the memory requirements of the model, while FLOPs specify the computational requirements

and efficiency, and finally, the inference time is a direct indicator of the real-time performance of (the maximum frame-per-seconds that can be obtained with) the model. We use data with a resolution of 240×180 to measure FLOPs and inference time, where the average inference times are calculated on a workstation with Quadro RTX 5000 GPU. We present the results of these computational complexity metrics in Table 7.13. Here, the numbers of model parameters are given in millions, FLOPs are given in billions (as GFLOPs), and the inference times are given in milliseconds. In this table, we use the same row for the methods that share the same deep architecture. Here, it can be seen that our method HyperE2VID provides a good trade-off between accuracy and efficiency. HyperE2VID is a significantly smaller and faster network than ET-Net while generating reconstructions with better visual quality. On the other hand, the smallest and fastest methods, FireNet and FireNet+, generate reconstructions with significantly lower visual quality.

Table 7.13 Computational complexity of network architectures in terms of the number of model parameters (in millions), number of floating point operations (FLOPS - in billions), and inference time (in milliseconds).

Network Architecture	Number of Params (M)	GFLOPs	Inference Time (ms)
E2VID [12, 101, 289]	10.71	20.07	<u>5.1</u>
FireNet [91, 289]	**0.04**	**1.62**	**1.6**
SPADE-E2VID [291]	11.46	68.06	16.1
ET-Net [112]	22.18	33.10	32.1
HyperE2VID [334]	<u>10.15</u>	<u>18.46</u>	6.6

7.8. Ablation Study

In the following ablation studies, we evaluate various design elements of the HyperE2VID model to verify their impact on performance. This includes a detailed comparison against the E2VID+ network [289], which shares similarities with our base network and employs the same training data. Specifically, we retrain E2VID+ with the same hyperparameters as HyperE2VID to assess the influence of these parameters independently of our hypernetwork architecture. We further investigate the role of previous reconstructions

by modifying the E2VID+ architecture to include them. Additionally, we compare our context-guided per-pixel dynamic convolutions with standard dynamic convolutions, confirming the superiority of our approach.

A significant part of our ablation study focuses on the use of context information. We experiment with networks using only event voxel grids as context, only previous reconstructions as context, or a combination of both, along with variations in curriculum learning and convolutional context fusion. We also investigate the effect of using different loss functions and validate our choices. Finally, we explore an alternative HyperE2VID architecture, employing sub-pixel convolutions [308] instead of bilinear upsampling in decoder blocks.

Training Settings. We retrained E2VID+ using the same setup and hyperparameters as HyperE2VID to test if E2VID+ could benefit from our hyperparameter choices, without our hypernetwork architecture. The results, shown in the second row of Table 7.14, reveal mixed outcomes. While the retrained E2VID+ shows improvements with respect to the original one in the ECD, ECD-Fast, MVSEC-Night, and FPVDR datasets, it falls short in the MVSEC and HQF datasets. This inconsistency suggests that the enhancements are not solely due to optimizing the hyperparameters. A direct comparison with HyperE2VID, under identical conditions, clearly shows the superiority of our hypernetworks-based approach.

Previous Reconstructions. In another experiment, we modify the E2VID+ architecture to include reconstructed intensity image from the previous timestep ($\hat{I}_{k-1}$) along with the current event tensor (V_k) via concatenation at the input. This is to distinguish the benefits of our architectural features from the simple use of past reconstructions. Even with the addition of curriculum learning, similar to HyperE2VID, this variant (shown in the third row of Table 7.14) underperforms compared to both the standard and retrained E2VID+. This highlights the unique effectiveness of our hypernetworks and dynamic per-pixel convolutions.

Table 7.14 Results from ablation experiments investigating effects of training settings, use of previous reconstructions, dynamic convolutions, and hypernetworks. The ECD-Fast and MVSEC-Night datasets introduced in Section 7.1.1. are denoted as Fast and Night, respectively.

	ECD			MVSEC			HQF			Fast	Night	FPVDR
	MSE	SSIM	LPIPS	MSE	SSIM	LPIPS	MSE	SSIM	LPIPS	BRISQUE		
E2VID+	0.070	0.503	0.236	0.132	0.262	0.514	0.036	**0.533**	**0.252**	22.627	12.285	18.677
E2VID+ (re-trained)	0.047	0.537	0.217	0.153	0.259	0.531	0.048	0.507	0.285	17.719	8.131	15.432
w/ $\hat{I}_{k-1}$ at input	0.077	0.479	0.259	0.226	0.218	0.567	0.037	0.496	0.270	18.830	10.983	21.176
w/ Dynamic Conv. [348]	0.060	0.503	0.246	0.119	0.270	0.493	**0.031**	0.529	**0.252**	15.162	6.602	20.758
w/ CondConv [347]	0.044	0.565	0.221	0.119	0.271	0.504	0.033	0.529	0.254	15.543	6.670	14.264
HyperE2VID	**0.033**	**0.576**	**0.212**	**0.076**	**0.315**	**0.476**	**0.031**	0.530	0.257	**14.024**	**5.973**	**14.178**

Table 7.15 Ablation results of HyperE2VID variants where we alter the context information, the existence of convolutional context fusion (CF) block, and curriculum learning (CL) strategy. The HQF-Slow, ECD-Fast, and MVSEC-Night datasets introduced in Section 7.1.1. are denoted as Slow, Fast, and Night, respectively.

			ECD			MVSEC			HQF			Slow			Fast	Night	FPVDR
Context	CL	CF	MSE	SSIM	LPIPS	MSE	SSIM	LPIPS	MSE	SSIM	LPIPS	MSE	SSIM	LPIPS	BRISQUE		
EVG			0.048	0.543	0.219	0.189	0.232	0.549	0.050	0.504	0.280	0.064	0.496	0.333	**13.42**	**1.05**	**4.66**
PR			0.050	0.536	0.229	0.181	0.228	0.573	0.035	0.517	0.276	0.039	0.558	0.283	18.79	10.12	16.18
EVG+PR			0.039	0.559	**0.212**	0.152	0.261	0.532	0.036	0.525	0.271	0.045	0.566	0.279	18.64	9.21	14.44
EVG+PR		✓	0.044	0.548	0.218	0.113	0.274	0.516	0.039	0.520	0.266	0.044	0.569	0.268	17.73	7.38	12.46
EVG+PR	✓		0.038	0.556	0.216	0.120	0.265	0.506	0.032	**0.534**	0.259	0.039	0.541	0.285	18.24	6.09	13.26
EVG+PR	✓	✓	**0.033**	**0.576**	**0.212**	**0.076**	**0.315**	**0.476**	**0.031**	0.530	**0.257**	**0.026**	**0.581**	**0.250**	14.02	5.97	14.18

Dynamic Convolutions. We also compare context-guided per-pixel dynamic convolutions with standard dynamic convolutions that lack these features. Training networks with dynamic convolutions [348] or CondConv [347] instead of the proposed CGDD block leads to a significant drop in performance, as shown in the fourth and fifth rows of Table 7.14. It highlights the effectiveness of the proposed context-guided per-pixel dynamic convolutions in HyperE2VID in enhancing reconstruction quality.

Context Information. Moreover, we carry out several ablation experiments in order to evaluate the design choices regarding the context information used for guiding the dynamic filter generation process in HyperE2VID. Specifically, we investigate hypernetworks that use only event voxel grids as context, only previous reconstructions as context, or a combination of both, denoted as EVG, PR, and EVG+PR, respectively. It should be emphasized that these HyperE2VID variants specifically modify the context tensor computation within the CF block, while maintaining the event tensor at the input of the head layer and preserving the dynamic network architecture of both the DFG and CGDD blocks. For EVG+PR, we also examine the impact of using the curriculum learning strategy (CL) and convolutional context fusion (CF). When CF is not used, we concatenate the previously reconstructed images and event tensors channel-wise and downsample the resulting tensor to match the input of the dynamic convolution in the CGDD block. The results are summarized in Table 7.15. Here, we also give the results on the HQF-Slow subset of the HQF dataset, containing slow motion.

Our quantitative results highlight the significance of choosing the right context based on the scene's characteristics. For instance, in slow-motion scenarios (HQF-Slow), the network utilizing solely previous reconstructions (PR) vastly outperforms the one using only event voxel grids as context (EVG). Conversely, in scenes with fast motion (ECD-Fast, FPVDR) or low light conditions (MVSEC-Night), PR's performance diminishes. To visually illustrate these findings, Figure 7.24 shows two representative scenes from our test datasets. The first scene, from the ECD-Fast segment of the ECD dataset, highlights the limitations of standard camera intensity frames for fast motion, which struggle with either motion blur or underexposure, while the event data adeptly captures the dynamic edges of the scene. This

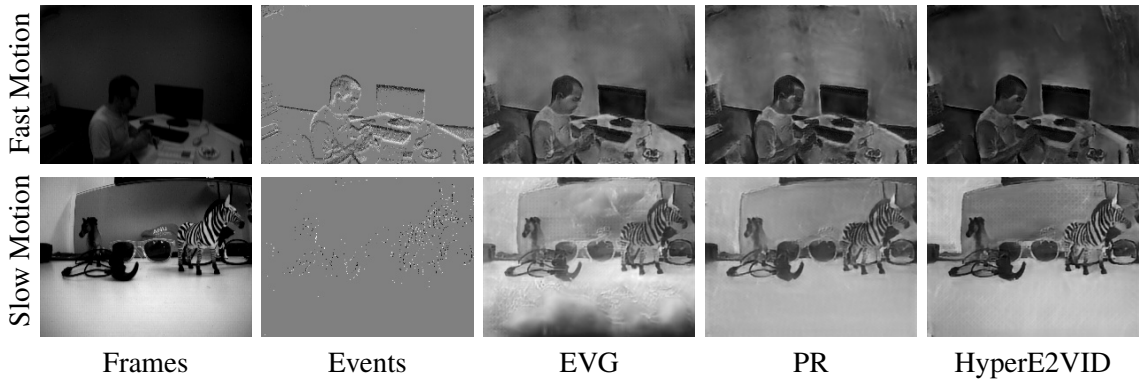


Figure 7.24 Understanding the role of context information. This figure shows frames, events, and reconstructions from two distinct scenes: one with fast motion (top) and another with slow motion (bottom). It highlights the significance of utilizing event and reconstruction data as context information for optimal results.

effectively demonstrates the strength of event data in high-speed conditions. The second scene, from the HQF-Slow segment of the HQF dataset, presents a slow-motion environment where intensity frames capture detailed visual information, but events are generated sparsely, capturing only significant brightness changes. Consequently, much of the visual detail in the scene is not visible in the event data.

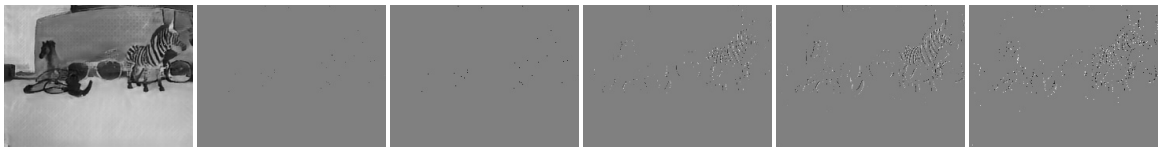
Our findings in Table 7.15 also reveal that leveraging both events and previous reconstructions as contextual information (EVG+PR) generally outperforms using only events (EVG) or only reconstructions (PR) as context. When using both events and reconstructions (EVG+PR), incorporating only the context fusion (CF) yields performance improvements on the MVSEC dataset. In contrast, incorporating only the curriculum learning strategy (CL) enhances performance on both the MVSEC and HQF datasets. Combining all these components results in our proposed HyperE2VID model (last row), which achieves the best scores on ECD, MVSEC, and HQF datasets, as well as the second-best scores on ECD-Fast and MVSEC-Night datasets. The variant using only event voxel grids as context (EVG), despite struggling on the HQF dataset and especially in its slow-motion sequences, excels in the fast-motion and night driving sequences. This is also visible in the top row of Figure 7.24, where the reconstruction of EVG is sharper and has minimal artifacts, even compared to the reconstruction of HyperE2VID. While HyperE2VID achieves the highest scores overall, the superior performance of the event-only model in

certain scenarios suggests potential room for improvement for our context fusion block for future work.

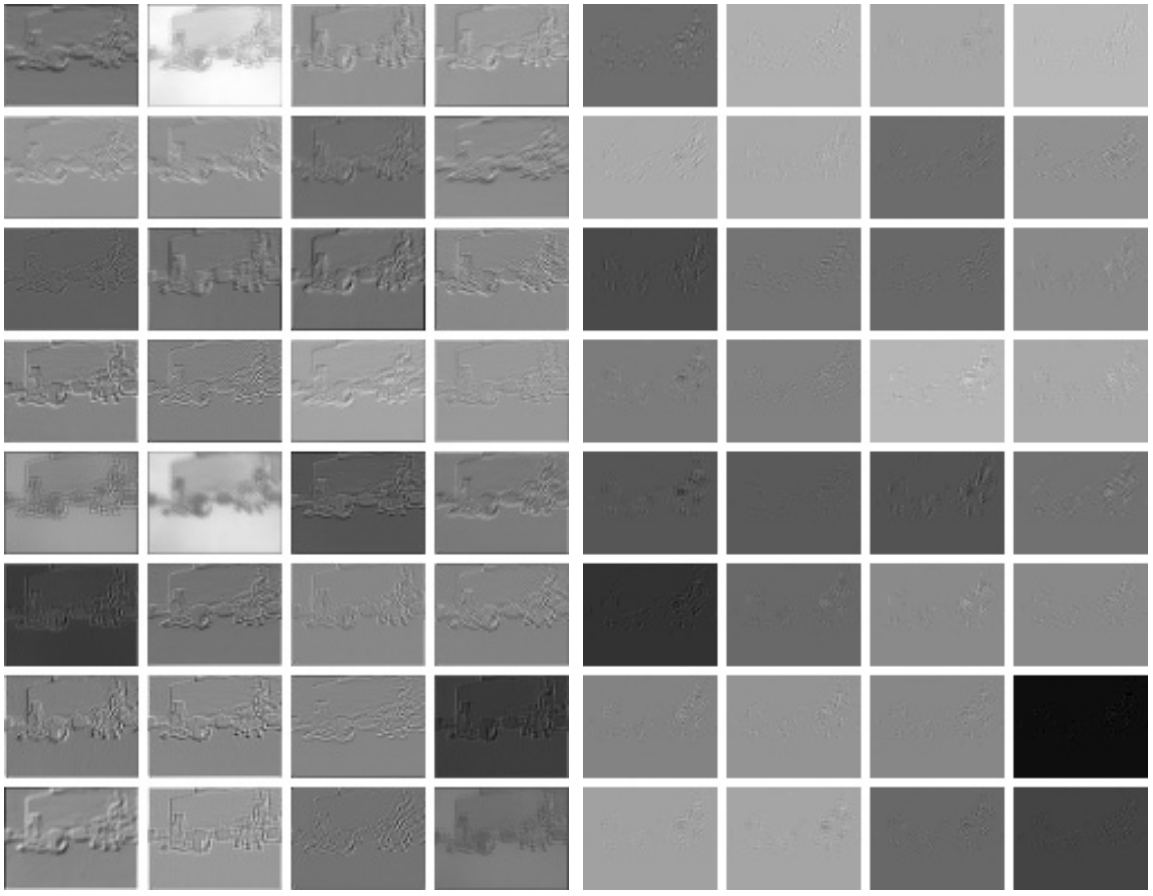
To further analyze the effect of context information, we present channel visualizations for each of the input and output tensors of the context fusion module in Figure 7.25, using a scene from the HQF-Slow subset of the HQF dataset. The top row shows the previous reconstruction (a) and the five temporal channels of the event tensor (b). Note that there is minimal visual information present in these event tensor channels due to the slow motion, while the previous reconstruction captures visual details from static parts. These form the inputs to our context fusion module. The output, a context tensor with all 32 channels (c), demonstrates various representations of static scene parts influenced by the previous reconstruction. For comparison, (d) visualizes the same channels using only the event tensor (using a zero tensor in place of $\hat{I}_{k-1}$). This ablation study reveals minimal visual information, reliant solely on the dynamic aspects captured by events.

Loss Function. To validate our choice of loss functions, we conduct additional ablation experiments comparing the performance of our method with and without the perceptual and temporal losses. For these experiments, we train our HyperE2VID network with L1 loss, instead of the perceptual and temporal losses. Moreover, we perform experiments where we employ either L1 or L2 loss instead of or alongside LPIPS. After training these models, we evaluate them on ECD, MVSEC, and HQF datasets and report their mean MSE, SSIM, and LPIPS scores. The results of these experiments, presented in Table 7.16, further prove the effectiveness of our chosen loss functions in enhancing the quality of reconstructed images.

Here, we observe that using L1 loss instead of the perceptual and temporal losses, although getting an MSE score close to that of HyperE2VID, underperforms according to SSIM and LPIPS metrics (row 2). Using L1 or L2 loss instead of LPIPS (and alongside temporal consistency loss) results in reconstructions with significantly lower perceptual similarity to ground truth images, as one might expect (rows 3 and 5). Using L2 loss instead of L1 loss generates reconstructions with lower structural similarity to ground truth images (rows 4 and 5 vs. rows 1, 2, and 3). Using L2 loss instead of LPIPS provides a better MSE score



(a) Prev. recon. (b) Each of the five channels (temporal bins) of the event voxel grid



(c) Channels of context tensor (d) Channels of context tensor w/o prev. recon.

Figure 7.25 Visualization of input and output tensors of context fusion module for an example scene from HQF-Slow (best viewed zoomed in). (a) The reconstruction that the network generated at the previous time step ($\hat{I}_{k-1}$) (b) Visualization of channels of event tensor, where each channel corresponds to a temporal bin in our voxel grid event representation. Note that there is little visual information present in these channels due to the slow motion in the scene. (c) Visualization of all 32 channels of the context tensor produced by the context fusion module in HyperE2VID, given the inputs from (a) and (b). Note that the context tensor contains various representations of the static parts of the scene, thanks to the previous reconstruction. (d) Visualization of the same channels when we only provide event tensor to context fusion, and using a zero tensor instead of $\hat{I}_{k-1}$, as an ablation. Here it can be seen that the context tensor contains minimal visual information, relying only on the dynamic parts of the scene captured by events.

Table 7.16 Results of ablation experiments on loss functions. We conduct experiments where we train the same network with different combinations of Temporal Consistency (TC) loss, Learned Perceptual Image Patch Similarity (LPIPS) loss, L1 loss, and L2 loss. We then compare each trained model according to their mean MSE, SSIM, and LPIPS results on the combination of ECD, MVSEC, and HQF datasets.

	Loss Functions				Test Metrics		
Training	TC	LPIPS	L1	L2	MSE	SSIM	LPIPS
Loss Ablation 1	✓	✓	✓		0.053	0.436	0.369
Loss Ablation 2			✓		0.055	0.431	0.395
Loss Ablation 3	✓		✓		0.056	**0.445**	0.441
Loss Ablation 4	✓	✓		✓	0.083	0.412	<u>0.367</u>
Loss Ablation 5	✓			✓	<u>0.051</u>	0.414	0.527
HyperE2VID	✓	✓			**0.050**	<u>0.443</u>	**0.346**

than other loss ablation experiments, as might be expected, but to our surprise, it is still slightly worse than the MSE score of HyperE2VID (row 5 vs. the last row). The scores of HyperE2VID, which is trained with a combination of perceptual loss and temporal loss, are presented in the last row. These scores are better overall than ablation variants trained with other loss combinations, demonstrating the effectiveness of our chosen loss functions.

Upsampling Mechanism. Here, we explore an alternative HyperE2VID architecture where we modify the upsampling mechanism in the decoder blocks of the network. Specifically, we employ sub-pixel convolution layers [308] with a stride of $1/2$ instead of bilinear upsampling, motivated by the fact that HyperE2VID occasionally exhibits a checkerboard pattern in its reconstructed images, especially in texture-less areas. We employ the commonly used efficient implementation of the sub-pixel convolution, where a standard convolutional layer with stride 1 generates an intermediate tensor with $C \times r^2$ channels, followed by a pixel shuffle operation that rearranges elements of this tensor so that the output has C channels, with r being the upsampling factor of the spatial dimensions ($r = 2$ in our case). We use the Leaky ReLU activation function after this convolution, with a slope of 0.01 for the negative valued inputs. We initialize the convolution weights according to the ICNR method from [369] to prevent checkerboard artifacts due to random initialization. Finally,

we utilize a filter after these convolutions, corresponding to the *approach B* proposed by Sugawara et al. in [370], to further diminish checkerboard artifacts.

We denote the resulting architecture as HyperE2VID_alt and train it using the same settings as the original HyperE2VID. The quantitative comparison of HyperE2VID and HyperE2VID_alt using full-reference and no-reference image quality metrics are given in Tables 7.17 and 7.18, respectively, while Figure 7.26 presents qualitative examples. A comparison between reconstructions of HyperE2VID_alt and HyperE2VID in Figure 7.26 demonstrates that the alternative architecture mostly eliminates the checkerboard artifacts, compared to the reconstructions of standard architecture with bilinear upsampling. However, HyperE2VID_alt also exhibits lower contrast, lower sharpness, and a few more artifacts and blemishes in some cases. The quantitative results with full-reference metrics presented in Table 7.17 show that HyperE2VID obtains better scores in standard benchmark sequences of ECD, MVSEC, and HQF, except for BS-ERGB where HyperE2VID_alt outperforms. On the other hand, no-reference metrics scores of HyperE2VID_alt given in Table 7.18 are mostly better than those of HyperE2VID, except for MANIQA metric.

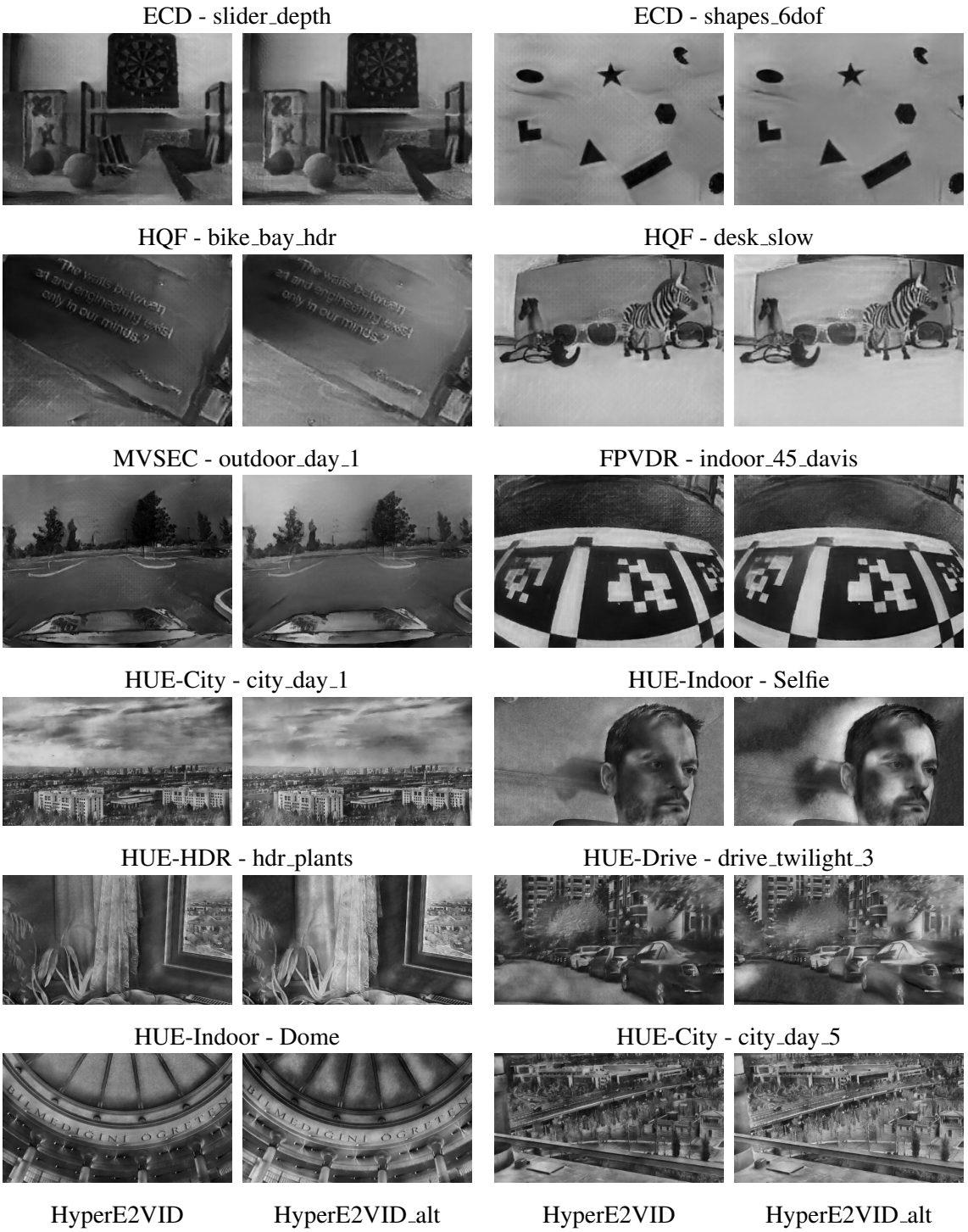


Figure 7.26 Qualitative comparison of HyperE2VID and HyperE2VID_alt. Although the alternative architecture largely mitigates checkerboard artifacts, it also displays reduced contrast and sharpness, alongside additional artifacts and blemishes in certain instances.

Table 7.17 Comparison of standard and alternative HyperE2VID models using full-reference quantitative results on the ECD, MVSEC, HQF, and BS-ERGB datasets. Here we use between-frames event grouping. Better score in each column is given in **bold**.

	ECD [280]			MVSEC [322]			HQF [289]			BS-ERGB [327]		
	MSE ↓	SSIM ↑	LPIPS ↓	MSE ↓	SSIM ↑	LPIPS ↓	MSE ↓	SSIM ↑	LPIPS ↓	MSE ↓	SSIM ↑	LPIPS ↓
HyperE2VID	**0.033**	**0.576**	**0.212**	**0.076**	**0.315**	**0.477**	**0.032**	**0.531**	0.261	0.077	0.360	0.446
HyperE2VID_alt	0.048	0.552	0.214	0.100	0.291	0.495	0.036	**0.531**	**0.259**	**0.071**	**0.370**	**0.439**

Table 7.18 Comparison of standard and alternative HyperE2VID models using no-reference quantitative results on challenging sequences of ECD-Fast, MVSEC-Night, HDR, and FPVDR. Here we use fixed-duration event grouping with a duration of 40 ms. Better score in each column is given in **bold**.

	ECD-Fast [280]			MVSEC-Night [322]			HDR [12]			FPVDR [328]		
	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑	BRISQUE ↓	NIQE ↓	MANIQA ↑
HyperE2VID	**14.215**	6.764	**0.281**	8.029	5.294	**0.344**	16.342	3.757	**0.345**	14.180	5.813	**0.286**
HyperE2VID_alt	15.288	**6.154**	0.260	**-0.218**	**5.002**	0.330	**10.875**	**3.557**	0.320	**10.422**	**4.934**	0.277

8. CONCLUSION

This final chapter concludes the thesis by providing a comprehensive summary and discussion, identifying current limitations, and suggesting potential paths for future research.

8.1. Summary

The past decade has seen significant progress in computer vision, driven by advancements in deep learning methodologies, leading to diverse applications across various domains. Despite these successes, artificial vision systems still lag behind their biological counterparts in tasks involving high-speed motion and real-time processing. Traditional frame-based sensors introduce challenges like motion blur and redundant information, hampering efficiency and real-time performance.

Event cameras, inspired by biological vision systems, offer promising solutions to these challenges. They operate asynchronously, generating events based on local intensity changes, resulting in high dynamic range, low latency, and minimal motion blur. This paradigm shift has sparked interest in event-based vision, necessitating novel processing methodologies to harness the unique characteristics of event data. We briefly reviewed the relevant literature in Chapter 2.

This thesis focused on **reconstructing intensity images from events**, leveraging their advantages for high-quality imaging in challenging scenarios involving low-light, high-speed, or high-dynamic range. Reconstruction enables the application of established methods developed for frame-based images and facilitates human-centered applications involving event data. It also serves as a bridge between event and frame-based modalities, proving useful for other tasks as well, such as simultaneous estimation of multiple quantities or facilitating unsupervised learning and domain adaptation.

Although recent methods have obtained impressive results in this task, the problem is still far from solved; considering the fact that state-of-the-art approaches use event representations

that cause latency, use computationally expensive models, and produce reconstructions that suffer from complications such as unrealistic artifacts. We discussed the limitations of existing methods in Chapter 3., and presented qualitative and quantitative results that display their failure cases in Chapter 7. Additionally, the evaluation setups in most studies possess several issues, like using datasets that are limited in scale and scope, focusing less on challenging scenarios where event cameras excel, overlooking the effect of some key variables that affect method performance, the absence of computational efficiency analysis, and shortcomings in openness and reproducibility. An overview of evaluation setups is presented in Section 5.1.2., while their limitations are discussed with more detail in Section 3.2.1. The issues in evaluation setups raise questions on the generalizability of the results to real-world scenarios and hinder fair comparison between methods.

Considering the limitations mentioned above, this thesis aimed to contribute to the literature on event-based video reconstruction by focusing on two main research objectives:

1. Having a better understanding of events and analyzing less explored computational methods to process them, to develop event-based video reconstruction methods surpassing existing ones in terms of both image quality and computational efficiency.

2. Evaluating these approaches in a unified and comprehensive manner to facilitate fair comparison and readiness to diverse real-world scenarios, by employing extensive real-world datasets, tackling challenging scenarios, acknowledging the influence of different variables, and assessing performance through multiple metrics and tasks.

For the first objective, we have proposed a novel dynamic neural network architecture based on hypernetworks, named **HyperE2VID**, in contrast to existing works that process the highly varying event data with static networks. Our method incorporated several innovative features in network design and training, including **per-pixel dynamic convolutions** that adapt to sparse and varying event data, a **context fusion** module that leverages complementary elements of event and frame domains, **filter decomposition** steps to reduce computational cost, and the application of **curriculum learning** to enhance training robustness. Chapter 6.

of this thesis delineated these key elements, while the experimental results in Chapter 7. demonstrated the importance of these design choices and the superiority of HyperE2VID in terms of both image quality and efficiency.

For the second objective, we introduced an open-source library for evaluating and analyzing event-based video reconstruction methods, **EVREAL**, and a new event dataset for assessing the quality of reconstructed images, **HUE**. With EVREAL, we addressed issues in existing evaluation setups, as described in Chapter 5. Specifically, EVREAL allowed us to extend the scope of evaluation with additional **datasets, metrics, and analysis settings** that target previously unreported challenges, such as scenarios involving rapid motion, low light, and high dynamic range. EVREAL also facilitates the analysis of method **robustness** across varied settings and conducts quantitative analysis on three **downstream tasks**, allowing a detailed assessment of each method's performance in relation to specific downstream objectives. Furthermore, our proposed dataset HUE enhanced the size and scope of existing test datasets thanks to its events with **high spatial resolution**, the **large number of sequences** taken in diverse scenarios, and a specific focus on **low-light** scenarios, which we detailed in Chapter 4.

In Chapter 7., we assessed HyperE2VID through comprehensive experiments and benchmarked it against existing methods. By employing EVREAL, we performed evaluations using multiple datasets, including our new dataset, HUE. We used a variety of image quality metrics and evaluated each method under various conditions. EVREAL also helped us assess reconstruction quality in downstream tasks, the robustness of methods with respect to varying factors such as event rate and temporal irregularity, and reconstruction quality in other cases such as reconstructing color images, high frame rate video reconstruction, and reconstruction during motionless event sequences. The results demonstrated the effectiveness of HyperE2VID across a broad range of settings. We also performed an analysis of computational complexity considering the number of model parameters, the number of floating point operations (FLOPs), and inference time. This analysis showed that our method provides a good trade-off between accuracy and efficiency. Then, an extensive ablation study was conducted to confirm the effectiveness of

the design decisions made for HyperE2VID. Specifically, we investigated the role of using previous reconstructions, dynamic convolutions, curriculum learning, context information, and loss functions. Finally, we explored an alternative HyperE2VID architecture, employing sub-pixel convolutions instead of bilinear upsampling in decoder blocks, and showed via experiments that this modification eliminates checkerboard artifacts.

8.2. Discussion

Event-based vision is a rapidly developing field. Since events differ significantly from typical frames, various event representations and processing methodologies are emerging in the literature, as reviewed in Chapter 2. Reconstructing intensity images from events is a multifaceted task within this field, encompassing diverse approaches for solving and evaluating it. As an emerging and complex task, it presents numerous challenges and opportunities for advancements.

There are two main motivations for reconstructing images from events. The first is for visualization purposes, particularly for human-centered applications of event data. The second motivation is to solve downstream tasks using the reconstructed images as an intermediate representation. For human-centered applications, reconstructions that are more appealing to human perception are desirable, while for the second motivation, it is essential to have reconstructions that enhance downstream task performance. These two goals may not always align perfectly, and advancements in one area might not necessarily translate to better solutions in the other, as demonstrated by the experimental results presented in Chapter 7. Therefore, it is important to have methods that are adaptive in nature and evaluation protocols that are aligned with the end goal.

For either case, event statistics play a crucial role in model performance. Event statistics are influenced by scene characteristics, motion, and camera parameters. Therefore, it is essential to consider these variables, as a method that performs well under specific settings may not be suitable for general use when these variables are expected to change. Methods either need to

accommodate these changes or be developed for and tested under the same settings as their intended end-use cases.

The recent trend in the literature, similar to various other domains, is the adoption of deep learning-based methods. While these methods have achieved impressive results in terms of image quality, they require increasingly more computations [112, 113, 291]. On the other hand, the characteristic features of event-based sensors include their asynchronous scene-dependent sampling mechanism and efficiency in terms of processed data. The ultimate promise of neuromorphic event-based systems lies in their success in challenging real-world scenarios, delivering real-time performance with energy efficiency. However, the trend of using larger and computationally expensive methods contradicts this direction.

In light of these observations, we argue that the two most crucial characteristics for event-based video reconstruction methods, besides image quality and particularly in their applicability to real-world scenarios, are generalizability and computational efficiency. Our proposed model, HyperE2VID, represents a solid step toward methods that are computationally cheaper, dynamically adaptive, and capable of producing higher-quality reconstructions. Our current work also has some limitations and areas for future development, which we discuss in Section 8.3. The key design components of HyperE2VID demonstrate the potential of dynamic network architectures and hypernetworks in processing highly variable event data, opening up new possibilities for future research in this direction and targeting additional tasks.

To demonstrate the generalizability of a method, diverse test settings and datasets are essential. Evaluation protocols require a unified pipeline for fair comparison and must be open for reproducibility. As a newly emerging field, there have been issues with the evaluation setups, as discussed in Chapter 5. Our proposed evaluation framework, EVREAL, is designed to address these issues. We complement it with a new dataset, HUE, which expands the scale and scope of existing benchmark datasets.

8.3. Limitations and Future Work

Similar to the majority of the works for event-based video reconstruction, we processed events in groups. This approach has the downside of introducing latency since each new frame is generated only when a new event group is formed, in contrast to works that process events as they arrive and can generate new frames anytime on demand, such as [124, 170]. On the other hand, HyperE2VID has the advantage of working successfully with various event grouping settings, thanks to its dynamic architecture. Therefore, if the computational requirements of performing frequent reconstructions are acceptable, one can use HyperE2VID with smaller temporal windows to reduce latency. As we have shown with Figure 7.22, HyperE2VID continues to generate high-quality outputs with event windows as short as $200\,\mu s$.

It is also possible to accumulate events for a longer time period and perform inference only when required to reduce computational load. However, this reduces the quality of reconstructions, as we demonstrate with experiments in Figure 7.21. This is expected since accumulating more events means discarding more information due to the aggregation mechanisms used in the event representation.

Event representation is another potential improvement area for event-based video reconstruction. Following the existing methods, we have focused on a specific voxel grid event representation with HyperE2VID. There are many other works regarding event representations in the literature (Section 2.2.1.), including the recently increased works on learned representations. Future work might explore other representations together with event-based video reconstruction methods, and training a learned representation together with HyperE2VID in an end-to-end manner has the potential for improvements.

While training HyperE2VID, we have only focused on the training set proposed in [289], generated with ESIM event simulator [294]. However, this dataset has weaknesses like constrained motion and unrealistic scenes. Furthermore, there are more recent simulators with improved event generation models [336, 371–374]. Training HyperE2VID with a

training dataset that is generated with more realistic scenes and event generation mechanisms can potentially improve it.

Another related element is the data augmentation used during training. While training HyperE2VID, we have used the data augmentation procedures proposed at [289]. These include a rather simple noise and pause augmentation. Using a better-modeled event noise might improve the realism of training events and the end results. Using noise events acquired from a real event camera might be another option, as presented in [375].

Our work demonstrated the potential of dynamic network architectures for processing highly varying event data and opens up possibilities for future research in this direction. Following our work, hypernetwork based architectures could be applied to other event-based vision tasks like optical-flow estimation as well. Furthermore, we used a simple context fusion block in HyperE2VID to guide dynamic filter generation, and we left the exploration of more sophisticated context fusion architectures for future work.

We used standard (*i.e.* non-spiking) ANNs for our work. Spiking neural networks are a natural fit to process event data, but they are harder to train. With the advancements in SNN training techniques, and the developments in spiking hardware allowing fast and efficient inference for SNNs, we expect to see more SNN based methods in the future.

We expect the event-based vision literature to keep growing and event cameras to become more prominent in the following years, with more applications in areas such as computational photography and robotics, enabling high-quality and robust imaging and perception in a fast and computationally efficient way. We believe our work represents a significant step in these directions.

REFERENCES

[1] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, **2004**.

[2] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595. **2018**.

[3] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *ECCV*, pages 170–185. **2018**.

[4] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.*, 21(12):4695–4708, **2012**.

[5] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212, **2012**.

[6] Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. MANIQA: Multi-dimension attention network for no-reference image quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1191–1200. **2022**.

[7] Eadweard Muybridge. Sallie gardner au galop. `https://commons.wikimedia.org/wiki/File:The_Horse_in_Motion_high_res.tiff`, **1878**.

[8] Yisa Zhang, Yuchen Zhao, Hengyi Lv, Yang Feng, Hailong Liu, and Chengshan Han. Adaptive slicing method of the spatiotemporal event stream obtained from a dynamic vision sensor. *Sensors*, 22(7):2614, **2022**.

[9] Daniel Gehrig, Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. Asynchronous, photometric feature tracking using events and frames. In *ECCV*, pages 750–765. **2018**.

[10] Guillermo Gallego. Event-based vision. `https://sites.google.com/view/guillermogallego/research/event-based-vision`, **2020**. [Online; accessed 17-April-2024].

[11] Elias Mueggler. *Event-based vision for high-speed robotics*. Ph.D. thesis, University of Zurich, **2017**.

[12] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6):1964–1980, **2019**.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, **2017**.

[14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, **2014**.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778. **2016**.

[16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788. **2016**.

[17] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, **2016**.

[18] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125. **2017**.

[19] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, pages 1907–1915. **2017**.

[20] KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5830–5840. **2021**.

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440. **2015**.

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969. **2017**.

[23] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2020**.

[24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026. **2023**.

[25] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, pages 1653–1660. **2014**.

[26] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499. **2016**.

[27] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, pages 7291–7299. **2017**.

[28] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, pages 7297–7306. **2018**.

[29] Ce Zheng, Wenhan Wu, Chen Chen, Taojiannan Yang, Sijie Zhu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah. Deep learning-based human pose estimation: A survey. *ACM Computing Surveys*, 56(1):1–37, **2023**.

[30] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *CVPR*, pages 8934–8943. **2018**.

[31] Nan Yang, Lukas von Stumberg, Rui Wang, and Daniel Cremers. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In *CVPR*, pages 1281–1292. **2020**.

[32] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *CVPR*, pages 5162–5170. **2015**.

[33] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, pages 270–279. **2017**.

[34] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, pages 2002–2011. **2018**.

[35] Yue Ming, Xuyang Meng, Chunxiao Fan, and Hui Yu. Deep learning for monocular depth estimation: A review. *Neurocomputing*, 438:14–33, **2021**.

[36] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *CVPR*, pages 4282–4291. **2019**.

[37] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, pages 2566–2576. **2019**.

[38] Jinkun Cao, Jiangmiao Pang, Xinshuo Weng, Rawal Khirodkar, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9686–9696. **2023**.

[39] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, pages 1725–1732. **2014**.

[40] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497. **2015**.

[41] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, **2017**.

[42] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, **2018**.

[43] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695. **2022**.

[44] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, **2015**.

[45] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, **1998**.

[46] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986. **2022**.

[47] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, **2020**.

[48] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022. **2021**.

[49] Daniel Lévy and Arzav Jain. Breast mass classification from mammograms using deep convolutional neural networks. *arXiv preprint arXiv:1612.00542*, **2016**.

[50] Sérgio Pereira, Adriano Pinto, Victor Alves, and Carlos A Silva. Brain tumor segmentation using convolutional neural networks in mri images. *IEEE transactions on medical imaging*, 35(5):1240–1251, **2016**.

[51] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, **2017**.

[52] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, **2017**.

[53] S Suganyadevi, V Seethalakshmi, and K Balasamy. A review on deep learning in medical image analysis. *International Journal of Multimedia Information Retrieval*, 11(1):19–38, **2022**.

[54] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, **2016**.

[55] Ashesh Jain, Avi Singh, Hema S Koppula, Shane Soh, and Ashutosh Saxena. Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In *ICRA*, pages 3118–3125. **2016**.

[56] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *NeurIPS*, pages 64–72. **2016**.

[57] Chelsea Finn, Tianhe Yu, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. One-shot visual imitation learning via meta-learning. *arXiv preprint arXiv:1709.04905*, **2017**.

[58] Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.*, 37(4-5):421–436, **2018**.

[59] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *NeurIPS*, pages 4967–4976. **2017**.

[60] Nicholas Watters, Daniel Zoran, Theophane Weber, Peter Battaglia, Razvan Pascanu, and Andrea Tacchetti. Visual interaction networks: Learning a physics simulator from video. In *NeurIPS*, pages 4539–4547. **2017**.

[61] Axel Sauer, Nikolay Savinov, and Andreas Geiger. Conditional affordance learning for driving in urban environments. *arXiv preprint arXiv:1806.06498*, **2018**.

[62] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *ICRA*, pages 8248–8254. **2019**.

[63] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *CoRL*, pages 66–75. **2020**.

[64] Lichao Mou, Pedram Ghamisi, and Xiao Xiang Zhu. Deep recurrent neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3639–3655, **2017**.

[65] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, **2013**.

[66] Sander Dieleman, Kyle W Willett, and Joni Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, 450(2):1441–1459, **2015**.

[67] E Neftci, C Posch, E Chicca, and H Ishibuchi. Neuromorphic engineering. *Computational Intelligence*, 2:278, **2015**.

[68] Shih-Chii Liu, Bodo Rueckauer, Enea Ceolini, Adrian Huber, and Tobi Delbruck. Event-driven sensing for efficient perception: Vision and audition algorithms. *IEEE Signal Processing Magazine*, 36(6):29–37, **2019**.

[69] Lichtsteiner Patrick, Christoph Posch, and Tobi Delbruck. A 128x 128 120 db 15μ s latency asynchronous temporal contrast vision sensor. *IEEE J. Solid-State Circuits*, 43:566–576, **2008**.

[70] Christoph Posch, Daniel Matolin, and Rainer Wohlgenannt. A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds. *IEEE Journal of Solid-State Circuits*, 46(1):259–275, **2010**.

[71] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE J. Solid-State Circuits*, 49(10):2333–2341, **2014**.

[72] Christoph Posch, Teresa Serrano-Gotarredona, Bernabe Linares-Barranco, and Tobi Delbruck. Retinomorphic event-based vision sensors: bioinspired cameras with spiking output. *Proceedings of the IEEE*, 102(10):1470–1484, **2014**.

[73] Massimo Antonio Sivilotti. *Wiring considerations in analog VLSI systems, with application to field-programmable networks*. California Institute of Technology, **1991**.

[74] Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J Davison, Jörg Conradt, Kostas Daniilidis, et al. Event-based vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(1):154–180, **2020**.

[75] Xu Zheng, Yexin Liu, Yunfan Lu, Tongyan Hua, Tianbo Pan, Weiming Zhang, Dacheng Tao, and Lin Wang. Deep learning for event-based vision: A comprehensive survey and benchmarks. *arXiv preprint arXiv:2302.08890*, **2023**.

[76] Nico Messikommer, Stamatios Georgoulis, Daniel Gehrig, Stepan Tulyakov, Julius Erbach, Alfredo Bochicchio, Yuanyou Li, and Davide Scaramuzza. Multi-bracket high dynamic range imaging with event cameras. In *CVPR*, pages 547–557. **2022**.

[77] Liyuan Pan, Richard Hartley, Cedric Scheerlinck, Miaomiao Liu, Xin Yu, and Yuchao Dai. High frame rate video reconstruction based on an event camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2020**.

[78] Lei Yu, Wen Yang, et al. Event-based high frame-rate video reconstruction with a novel cycle-event network. In *ICIP*, pages 86–90. **2020**.

[79] Liyuan Pan, Cedric Scheerlinck, Xin Yu, Richard Hartley, Miaomiao Liu, and Yuchao Dai. Bringing a blurry frame alive at high frame-rate with an event camera. In *CVPR*, pages 6820–6829. **2019**.

[80] Chen Haoyu, Teng Minggui, Shi Boxin, Wang YIzhou, and Huang Tiejun. Learning to deblur and generate high frame rate video with an event camera. *arXiv preprint arXiv:2003.00847*, **2020**.

[81] Chu Zhou, Minggui Teng, Jin Han, Jinxiu Liang, Chao Xu, Gang Cao, and Boxin Shi. Deblurring low-light images with events. *International Journal of Computer Vision*, 131(5):1284–1298, **2023**.

[82] Hongmin Li, Guoqi Li, Hanchao Liu, and Luping Shi. Super-resolution of spatiotemporal event-stream image captured by the asynchronous temporal contrast vision sensor. *arXiv preprint arXiv:1802.02398*, **2018**.

[83] Siqi Li, Yutong Feng, Yipeng Li, Yu Jiang, Changqing Zou, and Yue Gao. Event stream super-resolution via spatiotemporal constraint learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4480–4489. **2021**.

[84] Yunfan Lu, Zipeng Wang, Minjie Liu, Hongjian Wang, and Lin Wang. Learning spatial-temporal implicit neural representations for event-guided video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1557–1567. **2023**.

[85] Song Zhang, Yu Zhang, Zhe Jiang, Dongqing Zou, Jimmy Ren, and Bin Zhou. Learning to see in the dark with events. In *ECCV*, volume 1, page 2. **2020**.

[86] Jinxiu Liang, Yixin Yang, Boyu Li, Peiqi Duan, Yong Xu, and Boxin Shi. Coherent event guided low-light video enhancement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10615–10625. **2023**.

[87] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *CVPR*, pages 3857–3866. **2019**.

[88] Ruipeng Guo, Qianwan Yang, Andrew S Chang, Guorong Hu, Joseph Greene, Christopher V Gabel, Sixian You, and Lei Tian. Eventlfm: Event camera integrated fourier light field microscopy for ultrafast 3d imaging. *arXiv preprint arXiv:2310.00730*, **2023**.

[89] Shintaro Shiba, Friedhelm Hamann, Yoshimitsu Aoki, and Guillermo Gallego. Event-based background-oriented schlieren. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2023**.

[90] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Continuous-time intensity estimation using event cameras. In *ACCV*, pages 308–324. **2018**. doi:10.1007/978-3-030-20873-8_20.

[91] Cedric Scheerlinck, Henri Rebecq, Daniel Gehrig, Nick Barnes, Robert Mahony, and Davide Scaramuzza. Fast image reconstruction with an event camera. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 156–163. **2020**.

[92] Zihao W Wang, Peiqi Duan, Oliver Cossairt, Aggelos Katsaggelos, Tiejun Huang, and Boxin Shi. Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging. In *CVPR*, pages 1609–1619. **2020**.

[93] Jin Han, Chu Zhou, Peiqi Duan, Yehui Tang, Chang Xu, Chao Xu, Tiejun Huang, and Boxin Shi. Neuromorphic camera guided high dynamic range imaging. In *CVPR*, pages 1730–1739. **2020**.

[94] Weng Fei Low and Gim Hee Lee. Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18335–18346. **2023**.

[95] Manasi Muglikar, Mathias Gehrig, Daniel Gehrig, and Davide Scaramuzza. How to calibrate your event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1403–1409. **2021**.

[96] Matthew Cook, Luca Gugelmann, Florian Jug, Christoph Krautz, and Angelika Steger. Interacting maps for fast visual interpretation. In *IJCNN*, pages 770–776. **2011**.

[97] H Kim, A Handa, R Benosman, SH Ieng, and AJ Davison. Simultaneous mosaicing and tracking with an event camera. In *BMVC*. **2014**.

[98] Hanme Kim, Stefan Leutenegger, and Andrew J Davison. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *ECCV*, pages 349–364. **2016**.

[99] Patrick Bardow, Andrew J Davison, and Stefan Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *CVPR*, pages 884–892. **2016**.

[100] Guillermo Gallego, Christian Forster, Elias Mueggler, and Davide Scaramuzza. Event-based camera pose tracking using a generative event model. *arXiv preprint arXiv:1510.01972*, **2015**.

[101] Federico Paredes-Vallés and Guido CHE de Croon. Back to event basics: Self-supervised learning of image reconstruction for event cameras via photometric constancy. In *CVPR*, pages 3446–3455. **2021**.

[102] Hoonhee Cho, Hyeonseong Kim, Yujeong Chae, and Kuk-Jin Yoon. Label-free event-based object recognition via joint learning with image reconstruction from events. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19866–19877. **2023**.

[103] Zhaoning Sun, Nico Messikommer, Daniel Gehrig, and Davide Scaramuzza. Ess: Learning event-based semantic segmentation from still images. In *European Conference on Computer Vision*, pages 341–357. Springer, **2022**.

[104] Linglin Jing, Yiming Ding, Yunpeng Gao, Zhigang Wang, Xu Yan, Dong Wang, Gerald Schaefer, Hui Fang, Bin Zhao, and Xuelong Li. Hpl-ess: Hybrid pseudo-labeling for unsupervised event-based semantic segmentation. *arXiv preprint arXiv:2403.16788*, **2024**.

[105] Xu Zheng and Lin Wang. Eventdance: Unsupervised source-free cross-modal adaptation for event-based object recognition. *arXiv preprint arXiv:2403.14082*, **2024**.

[106] Chuyun Xie, Wei Gao, and Ren Guo. Cross-modal learning for event-based semantic segmentation via attention soft alignment. *IEEE Robotics and Automation Letters*, **2024**.

[107] Peiqi Duan, Zihao W Wang, Xinyu Zhou, Yi Ma, and Boxin Shi. Eventzoom: Learning to denoise and super resolve neuromorphic events. In *CVPR*, pages 12824–12833. **2021**.

[108] Soikat Hasan Ahmed, Hae Woong Jang, SM Nadim Uddin, and Yong Ju Jung. Deep event stereo leveraged by event-to-image translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 882–890. **2021**.

[109] Carlos Plou, Nerea Gallego, Alberto Sabater, Eduardo Montijano, Pablo Urcola, Luis Montesano, Ruben Martinez-Cantin, and Ana C Murillo. Eventsleep: Sleep activity recognition with event cameras. *arXiv preprint arXiv:2404.01801*, **2024**.

[110] Shafiq Ahmad, Pietro Morerio, and Alessio Del Bue. Person re-identification without identification via event anonymization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11132–11141. **2023**.

[111] Gemma Taverni, Diederik Paul Moeys, Chenghan Li, Celso Cavaco, Vasyl Motsnyi, David San Segundo Bello, and Tobi Delbruck. Front and back illuminated dynamic and active pixel vision sensors comparison. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 65(5):677–681, **2018**.

[112] Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Event-based video reconstruction using transformer. In *ICCV*, pages 2563–2572. **2021**.

[113] Quanmin Liang, Xiawu Zheng, Kai Huang, Yan Zhang, Jie Chen, and Yonghong Tian. Event-diffusion: Event-based image reconstruction and restoration with diffusion models. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3837–3846. **2023**.

[114] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, pages 740–755. **2014**.

[115] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, **2015**.

[116] Germain Haessig, Damien Joubert, Justin Haque, Moritz B Milde, Tobi Delbruck, and Viktor Gruev. Pdavis: Bio-inspired polarization event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3963–3972. **2023**.

[117] Tiejun Huang, Yajing Zheng, Zhaofei Yu, Rui Chen, Yuan Li, Ruiqin Xiong, Lei Ma, Junwei Zhao, Siwei Dong, Lin Zhu, et al. 1000× faster camera and machine vision with ordinary devices. *Engineering*, 25:110–119, **2023**.

[118] Evgeny V Votyakov and Alessandro Artusi. Quantifying noise of dynamic vision sensor. *arXiv preprint arXiv:2404.01948*, **2024**.

[119] Yuji Nozaki and Tobi Delbruck. Temperature and parasitic photocurrent effects in dynamic vision sensors. *IEEE Transactions on Electron Devices*, 64(8):3239–3245, **2017**.

[120] Christian Brandli, Lorenz Muller, and Tobi Delbruck. Real-time, high-speed video decompression using a frame-and event-based davis sensor. In *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 686–689. IEEE, **2014**.

[121] Ziwei Wang, Yonhon Ng, Pieter van Goor, and Robert Mahony. Event camera calibration of per-pixel biased contrast threshold. *arXiv preprint arXiv:2012.09378*, **2020**.

[122] Himanshu Akolkar, Cedric Meyer, Xavier Clady, Olivier Marre, Chiara Bartolozzi, Stefano Panzeri, and Ryad Benosman. What can neuromorphic event-driven precise timing add to spike-based pattern recognition? *Neural computation*, 27(3):561–593, **2015**.

[123] David Weikersdorfer and Jörg Conradt. Event-based particle filtering for robot self-localization. In *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 866–870. **2012**.

[124] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Continuous-time intensity estimation using event cameras. In *ACCV*, pages 308–324. **2018**.

[125] Alexander Kugele, Thomas Pfeil, Michael Pfeiffer, and Elisabetta Chicca. Efficient processing of spatio-temporal data streams with spiking neural networks. *Frontiers in Neuroscience*, 14:439, **2020**.

[126] Federico Paredes-Vallés, Kirk Yannick Willehm Scheper, and Guido Cornelis Henricus Eugene De Croon. Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2019**.

[127] Stefano Pini, Guido Borghi, Roberto Vezzani, and Rita Cucchiara. Video synthesis from intensity and event frames. In *ICIAP*, pages 313–323. **2019**.

[128] Ana I Maqueda, Antonio Loquercio, Guillermo Gallego, Narciso García, and Davide Scaramuzza. Event-based vision meets deep learning on steering prediction for self-driving cars. In *CVPR*, pages 5419–5427. **2018**.

[129] Jürgen Kogler, Christoph Sulzbachner, and Wilfried Kubinger. Bio-inspired stereo vision system with silicon retina imagers. In *ICVS*, pages 174–183. **2009**.

[130] Anton Mitrokhin, P Sutor, Cornelia Fermüller, and Yiannis Aloimonos. Learning sensorimotor control with neuromorphic sensors: Toward hyperdimensional active perception. *Science Robotics*, 4(30), **2019**.

[131] Xavier Lagorce, Garrick Orchard, Francesco Galluppi, Bertram E Shi, and Ryad B Benosman. Hots: a hierarchy of event-based time-surfaces for pattern recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(7):1346–1359, **2016**.

[132] Anton Mitrokhin, Cornelia Fermuller, Chethan Parameshwara, and Yiannis Aloimonos. Event-based moving object detection and tracking. *arXiv preprint arXiv:1803.04523*, **2018**.

[133] Yusuke Sekikawa, Kosuke Hara, and Hideo Saito. Eventnet: Asynchronous recursive event processing. In *CVPR*, pages 3887–3896. **2019**.

[134] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *CVPR*, pages 989–997. **2019**.

[135] David Tedaldi, Guillermo Gallego, Elias Mueggler, and Davide Scaramuzza. Feature detection and tracking with the dynamic and active-pixel vision sensor (davis). In *EBCCSP*, pages 1–7. **2016**.

[136] Beat Kueng, Elias Mueggler, Guillermo Gallego, and Davide Scaramuzza. Low-latency visual odometry using event-based feature tracks. In *IROS*, pages 16–23. **2016**.

[137] R Wes Baldwin, Ruixu Liu, Mohammed Almatrafi, Vijayan Asari, and Keigo Hirakawa. Time-ordered recent event (tore) volumes for event cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2519–2532, **2022**.

[138] Fuqiang Gu, Yong Lee, Yuan Zhuang, You Li, Jingbin Liu, Fangwen Yu, Ruiyuan Li, and Chao Chen. Mdoe: A spatiotemporal event representation considering the magnitude and density of events. *IEEE Robotics and Automation Letters*, 7(3):7966–7973, **2022**.

[139] Jianhao Jiao, Huaiyang Huang, Liang Li, Zhijian He, Yilong Zhu, and Ming Liu. Comparing representations in tracking for event camera-based slam. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 1369–1376. **2021**.

[140] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Ev-flownet: Self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*, **2018**.

[141] Chengxi Ye, Anton Mitrokhin, Cornelia Fermüller, James A Yorke, and Yiannis Aloimonos. Unsupervised learning of dense optical flow, depth and egomotion from sparse event data. *arXiv preprint arXiv:1809.08625*, **2018**.

[142] Daniel Gehrig, Antonio Loquercio, Konstantinos G Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *ICCV*, pages 5633–5643. **2019**.

[143] Marco Cannici, Marco Ciccone, Andrea Romanoni, and Matteo Matteucci. A differentiable recurrent surface for asynchronous event-based data. In *ECCV*, pages 1–17. **2020**.

[144] Dongsheng Wang, Xu Jia, Yang Zhang, Xinyu Zhang, Yaoyuan Wang, Ziyang Zhang, Dong Wang, and Huchuan Lu. Dual memory aggregation network for event-based object detection with learnable representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2492–2500. **2023**.

[145] Lakshmi Annamalai, Vignesh Ramanathan, and Chetan Singh Thakur. Event-lstm: An unsupervised and asynchronous learning-based representation for event-based data. *IEEE Robotics and Automation Letters*, 7(2):4678–4685, **2022**.

[146] Minggui Teng, Chu Zhou, Hanyue Lou, and Boxin Shi. Nest: Neural event stack for event-based image enhancement. In *European Conference on Computer Vision*, pages 660–676. Springer, **2022**.

[147] Nikola Zubić, Daniel Gehrig, Mathias Gehrig, and Davide Scaramuzza. From chaos comes order: Ordering event representations for object recognition and detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12846–12856. **2023**.

[148] Min Liu and Tobi Delbruck. Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors. In *BMVC*. **2018**.

[149] Jianing Li, Jia Li, Lin Zhu, Xijie Xiang, Tiejun Huang, and Yonghong Tian. Asynchronous spatio-temporal memory network for continuous event-based object detection. *IEEE Trans. Image Process.*, 31:2975–2987, **2022**.

[150] Urbano Miguel Nunes, Ryad Benosman, and Sio-Hoi Ieng. Adaptive global decay process for event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9771–9780. **2023**.

[151] Zhaoxuan Guo, Jiandong Gao, Guangyuan Ma, and Jiangtao Xu. Spatio-temporal aggregation transformer for object detection with neuromorphic vision sensors. *IEEE Sensors Journal*, **2024**.

[152] Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad Benosman. Hats: Histograms of averaged time surfaces for robust event-based object classification. In *CVPR*, pages 1731–1740. **2018**.

[153] Yongjian Deng, Youfu Li, and Hao Chen. Amae: Adaptive motion-agnostic encoder for event-based object classification. *IEEE Robotics and Automation Letters*, 5(3):4596–4603, **2020**.

[154] Tobias Brosch, Stephan Tschechne, and Heiko Neumann. On event-based optical flow detection. *Frontiers in neuroscience*, 9:137, **2015**.

[155] Jun Haeng Lee, Tobi Delbruck, and Michael Pfeiffer. Training deep spiking neural networks using backpropagation. *Frontiers in neuroscience*, 10:508, **2016**.

[156] Emre O Neftci, Charles Augustine, Somnath Paul, and Georgios Detorakis. Event-driven random back-propagation: Enabling neuromorphic deep learning machines. *Frontiers in neuroscience*, 11:324, **2017**.

[157] Dongsung Huh and Terrence J Sejnowski. Gradient descent for spiking neural networks. In *NeurIPS*, pages 1433–1443. **2018**.

[158] Yujie Wu, Lei Deng, Guoqi Li, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12:323875, **2018**.

[159] Qiaoyi Su, Yuhong Chou, Yifan Hu, Jianing Li, Shijie Mei, Ziyang Zhang, and Guoqi Li. Deep directly-trained spiking neural networks for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6555–6565. **2023**.

[160] Mathias Gehrig, Sumit Bam Shrestha, Daniel Mouritzen, and Davide Scaramuzza. Event-based angular velocity regression with spiking networks. *arXiv preprint arXiv:2003.02790*, **2020**.

[161] José Antonio Pérez-Carrasco, Bo Zhao, Carmen Serrano, Begona Acha, Teresa Serrano-Gotarredona, Shouchun Chen, and Bernabé Linares-Barranco. Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing–application to feedforward convnets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(11):2706–2719, **2013**.

[162] Yongqiang Cao, Yang Chen, and Deepak Khosla. Spiking deep convolutional neural networks for energy-efficient object recognition. *International Journal of Computer Vision*, 113(1):54–66, **2015**.

[163] Daniel Neil, Michael Pfeiffer, and Shih-Chii Liu. Learning to be efficient: Algorithms for training low-latency, low-compute deep spiking neural networks. In *Proceedings of the 31st annual ACM symposium on applied computing*, pages 293–298. **2016**.

[164] Evangelos Stromatias, Miguel Soto, Teresa Serrano-Gotarredona, and Bernabé Linares-Barranco. An event-driven classifier for spiking neural networks fed with synthetic or dynamic vision sensor data. *Frontiers in neuroscience*, 11:350, **2017**.

[165] Bodo Rueckauer, Iulia-Alexandra Lungu, Yuhuang Hu, Michael Pfeiffer, and Shih-Chii Liu. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in neuroscience*, 11:682, **2017**.

[166] Paul A Merolla, John V Arthur, Rodrigo Alvarez-Icaza, Andrew S Cassidy, Jun Sawada, Filipp Akopyan, Bryan L Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668–673, **2014**.

[167] Chi-Sang Poon and Kuan Zhou. Neuromorphic silicon neurons and large-scale neural networks: challenges and opportunities. *Frontiers in neuroscience*, 5:12834, **2011**.

[168] Chankyu Lee, Adarsh Kumar Kosta, Alex Zihao Zhu, Kenneth Chaney, Kostas Daniilidis, and Kaushik Roy. Spike-flownet: event-based optical flow estimation with energy-efficient hybrid neural networks. In *European Conference on Computer Vision*, pages 366–382. Springer, **2020**.

[169] Sourav Sanyal, Rohan Kumar Manna, and Kaushik Roy. Ev-planner: Energy-efficient robot navigation via event-based physics-guided neuromorphic planner. *IEEE Robotics and Automation Letters*, **2024**.

[170] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Asynchronous spatial image convolutions for event cameras. *IEEE Robotics and Automation Letters*, 4(2):816–822, **2019**.

[171] Marco Cannici, Marco Ciccone, Andrea Romanoni, and Matteo Matteucci. Asynchronous convolutional networks for object detection in neuromorphic cameras. In *CVPRW*. **2019**.

[172] Nico Messikommer, Daniel Gehrig, Antonio Loquercio, and Davide Scaramuzza. Event-based asynchronous sparse convolutional networks. *arXiv preprint arXiv:2003.09148*, **2020**.

[173] Leandro de Souza Rosa, Aiko Dinale, Simeon Bamford, Chiara Bartolozzi, and Arren Glover. High-throughput asynchronous convolutions for high-resolution event-cameras. In *2022 8th International Conference on Event-Based Control, Communication, and Signal Processing (EBCCSP)*, pages 1–8. IEEE, **2022**.

[174] Yijin Li, Han Zhou, Bangbang Yang, Ye Zhang, Zhaopeng Cui, Hujun Bao, and Guofeng Zhang. Graph-based asynchronous event processing for rapid object recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 934–943. **2021**.

[175] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *NeurIPS*, pages 6571–6583. **2018**.

[176] Giorgio Giannone, Asha Anoosheh, Alessio Quaglino, Pierluca D'Oro, Marco Gallieri, and Jonathan Masci. Real-time classification from short event-camera streams using input-filtering neural odes. *arXiv preprint arXiv:2004.03156*, **2020**.

[177] Valentina Vasco, Arren Glover, and Chiara Bartolozzi. Fast event-based harris corner detection exploiting the advantages of event-driven cameras. In *IROS*, pages 4144–4149. **2016**.

[178] Elias Mueggler, Chiara Bartolozzi, and Davide Scaramuzza. Fast event-based corner detection. In *BMVC*. **2017**.

[179] Ignacio Alzugaray and Margarita Chli. Asynchronous corner detection and tracking for event cameras in real time. *IEEE Robot. Autom. Lett.*, 3(4):3177–3184, **2018**.

[180] Bharath Ramesh, Andrés Ussa, Luca Della Vedova, Hong Yang, and Garrick Orchard. Low-power dynamic object detection and classification with freely moving event cameras. *Frontiers in Neuroscience*, 14:135, **2020**.

[181] Nitin J Sanket, Chethan M Parameshwara, Chahat Deep Singh, Ashwin V Kuruttukulam, Cornelia Fermüller, Davide Scaramuzza, and Yiannis Aloimonos. Evdodgenet: Deep dynamic obstacle dodging with event cameras. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10651–10657. IEEE, **2020**.

[182] Etienne Perot, Pierre de Tournemire, Davide Nitti, Jonathan Masci, and Amos Sironi. Learning to detect objects with a 1 megapixel event camera. *arXiv preprint arXiv:2009.13436*, **2020**.

[183] Anthony Bisulco, Fernando Cladera Ojeda, Volkan Isler, and Daniel D Lee. Fast motion understanding with spatiotemporal neural networks and dynamic vision sensors. *arXiv preprint arXiv:2011.09427*, **2020**.

[184] Daniel Neil, Michael Pfeiffer, and Shih-Chii Liu. Phased lstm: Accelerating recurrent network training for long or event-based sequences. *NeurIPS*, 29:3882–3890, **2016**.

[185] Javier Hidalgo-Carrió, Daniel Gehrig, and Davide Scaramuzza. Learning monocular dense depth from events. *arXiv preprint arXiv:2010.08350*, **2020**.

[186] Alex Zihao Zhu, Ziyun Wang, Kaung Khant, and Kostas Daniilidis. Eventgan: Leveraging large scale image datasets for event cameras. *arXiv preprint arXiv:1912.01584*, **2019**.

[187] Lin Wang, S. Mohammad Mostafavi, Yo-Sung Ho, Kuk-Jin Yoon, et al. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *CVPR*, pages 10081–10090. **2019**.

[188] Yin Bi, Aaron Chadha, Alhabib Abbas, Eirina Bourtsoulatze, and Yiannis Andreopoulos. Graph-based object classification for neuromorphic vision sensing. In *ICCV*, pages 491–501. **2019**.

[189] Anton Mitrokhin, Zhiyuan Hua, Cornelia Fermuller, and Yiannis Aloimonos. Learning visual motion segmentation using event surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14414–14423. **2020**.

[190] Yongjian Deng, Hao Chen, Hai Liu, and Youfu Li. A voxel graph cnn for object classification with event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1172–1181. **2022**.

[191] Alberto Sabater, Luis Montesano, and Ana C Murillo. Event transformer. a sparse-aware solution for efficient event data processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2677–2686. **2022**.

[192] Alberto Sabater, Luis Montesano, and Ana C Murillo. Event transformer+. a multi-purpose solution for efficient event data processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2023**.

[193] Yansong Peng, Yueyi Zhang, Zhiwei Xiong, Xiaoyan Sun, and Feng Wu. Get: group event transformer for event-based vision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6038–6048. **2023**.

[194] Mathias Gehrig and Davide Scaramuzza. Recurrent vision transformers for object detection with event cameras. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13884–13893. **2023**.

[195] Yansong Peng, Hebei Li, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. Scene adaptive sparse transformer for event-based object detection. *arXiv preprint arXiv:2404.01882*, **2024**.

[196] Pei Wang, Jiumei He, Qingsen Yan, Yu Zhu, Jinqiu Sun, and Yanning Zhang. Diffevent: Event residual diffusion for image deblurring. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3450–3454. IEEE, **2024**.

[197] Nikola Zubić, Mathias Gehrig, and Davide Scaramuzza. State space models for event cameras. *arXiv preprint arXiv:2402.15584*, **2024**.

[198] Ju Huang, Shiao Wang, Shuai Wang, Zhe Wu, Xiao Wang, and Bo Jiang. Mamba-fetrack: Frame-event tracking via state space model. *arXiv preprint arXiv:2404.18174*, **2024**.

[199] Marco Cannici, Marco Ciccone, Andrea Romanoni, and Matteo Matteucci. Attention mechanisms for object recognition with event-based cameras. In *WACV*, pages 1127–1136. **2019**.

[200] Junho Kim, Jaehyeok Bae, Gangin Park, Dongsu Zhang, and Young Min Kim. N-imagenet: Towards robust, fine-grained object recognition with event cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2146–2156. **2021**.

[201] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based optical flow using motion compensation. In *ECCV*. **2018**.

[202] Liyuan Pan, Miaomiao Liu, and Richard Hartley. Single image optical flow estimation with an event camera. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1669–1678. IEEE, **2020**.

[203] Mohammed Mutlaq Almatrafi, Raymond Baldwin, Kiyoharu Aizawa, and Keigo Hirakawa. Distance surface for event-based optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2020**.

[204] Jesse Hagenaars, Federico Paredes-Vallés, and Guido De Croon. Self-supervised learning of event-based optical flow with spiking neural networks. *Advances in Neural Information Processing Systems*, 34:7167–7179, **2021**.

[205] Mathias Gehrig, Mario Millhäusler, Daniel Gehrig, and Davide Scaramuzza. E-raft: Dense optical flow from event cameras. In *2021 International Conference on 3D Vision (3DV)*, pages 197–206. IEEE, **2021**.

[206] Zhexiong Wan, Yuchao Dai, and Yuxin Mao. Learning dense and continuous optical flow from an event camera. *IEEE Transactions on Image Processing*, 31:7237–7251, **2022**.

[207] Mathias Gehrig, Manasi Muglikar, and Davide Scaramuzza. Dense continuous-time optical flow from events and frames. *arXiv preprint arXiv:2203.13674*, **2022**.

[208] Shintaro Shiba, Yannick Klose, Yoshimitsu Aoki, and Guillermo Gallego. Secrets of event-based optical flow, depth and ego-motion estimation by contrast maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2024**.

[209] Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. In *British Machine Vision Conference*. **2017**.

[210] Alex Zihao Zhu, Nikolay Atanasov, and Kostas Daniilidis. Event-based visual inertial odometry. In *CVPR*, pages 5391–5399. **2017**.

[211] Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios. *IEEE Robot. Autom. Lett.*, 3(2):994–1001, **2018**.

[212] Elias Mueggler, Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. Continuous-time visual-inertial odometry for event cameras. *IEEE Transactions on Robotics*, 34(6):1425–1440, **2018**.

[213] Anh Nguyen, Thanh-Toan Do, Darwin G Caldwell, and Nikos G Tsagarakis. Real-time 6dof pose relocalization for event cameras with stacked spatial lstm networks. In *CVPRW*. **2019**.

[214] Dekai Zhu, Zhongcong Xu, Jinhu Dong, Canbo Ye, Yinbai Hu, Hang Su, Zhengfa Liu, and Guang Chen. Neuromorphic visual odometry system for intelligent vehicle application with bio-inspired vision sensor. In *ROBIO*, pages 2225–2232. **2019**.

[215] Daqi Liu, Alvaro Parra, and Tat-Jun Chin. Spatiotemporal registration for event-based visual odometry. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4937–4946. **2021**.

[216] Javier Hidalgo-Carrió, Guillermo Gallego, and Davide Scaramuzza. Event-aided direct sparse odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5781–5790. **2022**.

[217] Simon Klenk, Marvin Motzet, Lukas Koestler, and Daniel Cremers. Deep event visual odometry. *arXiv preprint arXiv:2312.09800*, **2023**.

[218] Bharath Ramesh, Shihao Zhang, Zhi Wei Lee, Zhi Gao, Garrick Orchard, and Cheng Xiang. Long-term object tracking with a moving event camera. In *BMVC*, page 241. **2018**.

[219] Francisco Barranco, Cornelia Fermuller, and Eduardo Ros. Real-time clustering and multi-target tracking using event-based sensors. In *IROS*, pages 5764–5769. **2018**.

[220] Jiqing Zhang, Xin Yang, Yingkai Fu, Xiaopeng Wei, Baocai Yin, and Bo Dong. Object tracking by jointly exploiting frame and event domain. In *ICCV*, pages 13043–13052. **2021**.

[221] Yajing Zheng, Zhaofei Yu, Song Wang, and Tiejun Huang. Spike-based motion estimation for object tracking through bio-inspired unsupervised learning. *IEEE Transactions on Image Processing*, 32:335–349, **2022**.

[222] Zhiyu Zhu, Junhui Hou, and Xianqiang Lyu. Learning graph-embedded key-event back-tracing for object tracking in event clouds. *Advances in Neural Information Processing Systems*, 35:7462–7476, **2022**.

[223] Jiqing Zhang, Bo Dong, Haiwei Zhang, Jianchuan Ding, Felix Heide, Baocai Yin, and Xin Yang. Spiking transformers for event-based single object tracking. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 8801–8810. **2022**.

[224] Xiao Wang, Jianing Li, Lin Zhu, Zhipeng Zhang, Zhe Chen, Xin Li, Yaowei Wang, Yonghong Tian, and Feng Wu. Visevent: Reliable object tracking via collaboration of frame and event flows. *IEEE Transactions on Cybernetics*, **2023**.

[225] Jens Egholm Pedersen, Raghav Singhal, and Jorg Conradt. Translation and scale invariance for event-based object tracking. In *Proceedings of the 2023 Annual Neuro-Inspired Computational Elements Conference*, pages 79–85. **2023**.

[226] Alex Zihao Zhu, Nikolay Atanasov, and Kostas Daniilidis. Event-based feature tracking with probabilistic data association. In *ICRA*, pages 4465–4470. **2017**.

[227] Jacques Manderscheid, Amos Sironi, Nicolas Bourdis, Davide Migliore, and Vincent Lepetit. Speed invariant time surface for learning to detect corner points with event-based cameras. In *CVPR*, pages 10245–10254. **2019**.

[228] Daniel Gehrig, Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. Eklt: Asynchronous photometric feature tracking using events and frames. *Int. J. Comput. Vis.*, 128(3):601–618, **2020**.

[229] Nico Messikommer, Carter Fang, Mathias Gehrig, and Davide Scaramuzza. Data-driven feature tracking for event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5642–5651. **2023**.

[230] Mohsen Firouzi and Jörg Conradt. Asynchronous event-based cooperative stereo matching using neuromorphic silicon retinas. *Neural Processing Letters*, 43:311–326, **2016**.

[231] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza. Semi-dense 3d reconstruction with a stereo event camera. In *ECCV*, pages 235–251. **2018**.

[232] Alexander Andreopoulos, Hirak J Kashyap, Tapan K Nayak, Arnon Amir, and Myron D Flickner. A low power, high throughput, fully event-based stereo system. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7532–7542. **2018**.

[233] Alex Zihao Zhu, Yibo Chen, and Kostas Daniilidis. Realtime time synchronized event-based stereo. In *ECCV*, pages 438–452. **2018**.

[234] Stepan Tulyakov, Francois Fleuret, Martin Kiefel, Peter Gehler, and Michael Hirsch. Learning an event sequence embedding for dense event-based deep stereo. In *ICCV*, pages 1527–1537. **2019**.

[235] Yixuan Wang, Jianing Li, Lin Zhu, Xijie Xiang, Tiejun Huang, and Yonghong Tian. Learning stereo depth estimation with bio-inspired spike cameras. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, **2022**.

[236] Xihao Chen, Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Depth from asymmetric frame-event stereo: A divide-and-conquer approach. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3045–3054. **2024**.

[237] Anton Mitrokhin, Chengxi Ye, Cornelia Fermüller, Yiannis Aloimonos, and Tobi Delbruck. Ev-imo: Motion segmentation dataset and learning pipeline for event cameras. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6105–6112. IEEE, **2019**.

[238] Timo Stoffregen, Guillermo Gallego, Tom Drummond, Lindsay Kleeman, and Davide Scaramuzza. Event-based motion segmentation by motion compensation. In *ICCV*, pages 7244–7253. **2019**.

[239] Chethan M Parameshwara, Nitin J Sanket, Arjun Gupta, Cornelia Fermuller, and Yiannis Aloimonos. Moms with events: Multi-object motion segmentation with monocular event cameras. *arXiv preprint arXiv:2006.06158*, **2020**.

[240] Yi Zhou, Guillermo Gallego, Xiuyuan Lu, Siqi Liu, and Shaojie Shen. Event-based motion segmentation with spatio-temporal graph cuts. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8):4868–4880, **2021**.

[241] Ziyun Wang, Jinyuan Guo, and Kostas Daniilidis. Un-evmoseg: Unsupervised event-based independent motion segmentation. *arXiv preprint arXiv:2312.00114*, **2023**.

[242] Inigo Alonso and Ana C Murillo. Ev-segnet: Semantic segmentation for event-based cameras. in 2019 ieee. In *CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1624–1633. **2019**.

[243] Zhiwen Chen, Zhiyu Zhu, Yifan Zhang, Junhui Hou, Guangming Shi, and Jinjian Wu. Segment any events via weighted adaptation of pivotal tokens. *arXiv preprint arXiv:2312.16222*, **2023**.

[244] Lingdong Kong, Youquan Liu, Lai Xing Ng, Benoit R Cottereau, and Wei Tsang Ooi. Openess: Event-based semantic scene understanding with open vocabularies. *arXiv preprint arXiv:2405.05259*, **2024**.

[245] Diederik Paul Moeys, Federico Corradi, Emmett Kerr, Philip Vance, Gautham Das, Daniel Neil, Dermot Kerr, and Tobi Delbrück. Steering a predator robot using a mixed frame/event-driven convolutional neural network. In *2016 Second international conference on event-based control, communication, and signal processing (EBCCSP)*, pages 1–8. IEEE, **2016**.

[246] Lea Steffen, Benedict Hauck, Jacques Kaiser, Jakob Weinland, Stefan Ulbrich, Daniel Reichard, Arne Roennau, and Rüdiger Dillmann. Creating an obstacle memory through event-based stereo vision and robotic proprioception. In *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, pages 1829–1836. IEEE, **2019**.

[247] Rajkumar Muthusamy, Xiaoqian Huang, Yahya Zweiri, Lakmal Seneviratne, and Dongming Gan. Neuromorphic event-based slip detection and suppression in robotic grasping and manipulation. *IEEE Access*, 8:153364–153384, **2020**.

[248] Juan Pablo Rodríguez-Gómez, Raul Tapia, Maria del Mar Guzmán Garcia, Jose Ramiro Martínez-de Dios, and Anibal Ollero. Free as a bird: Event-based dynamic sense-and-avoid for ornithopter robot flight. *IEEE Robot. Autom. Lett.*, 7(2):5413–5420, **2022**.

[249] Abdulla Ayyad, Mohamad Halwani, Dewald Swart, Rajkumar Muthusamy, Fahad Almaskari, and Yahya Zweiri. Neuromorphic vision based control for the precise positioning of robotic drilling systems. *Robotics and Computer-Integrated Manufacturing*, 79:102419, **2023**.

[250] Benedek Forrai, Takahiro Miki, Daniel Gehrig, Marco Hutter, and Davide Scaramuzza. Event-based agile object catching with a quadrupedal robot. *arXiv preprint arXiv:2303.17479*, **2023**.

[251] Guang Chen, Lin Hong, Jinhu Dong, Peigen Liu, Jörg Conradt, and Alois Knoll. Eddd: Event-based drowsiness driving detection through facial motion analysis with neuromorphic vision sensor. *IEEE Sensors Journal*, 20(11):6170–6181, **2020**.

[252] Chu Yang, Peigen Liu, Guang Chen, Zhengfa Liu, Ya Wu, and Alois Knoll. Event-based driver distraction detection and action recognition. In *2022 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–7. IEEE, **2022**.

[253] Waseem Shariff, Mehdi Sefidgar Dilmaghani, Paul Kielty, Joe Lemley, Muhammad Ali Farooq, Faisal Khan, and Peter Corcoran. Neuromorphic driver monitoring systems: A computationally efficient proof-of-concept for driver distraction detection. *IEEE Open Journal of Vehicular Technology*, **2023**.

[254] Paul Kielty, Mehdi Sefidgar Dilmaghani, Waseem Shariff, Cian Ryan, Joe Lemley, and Peter Corcoran. Neuromorphic driver monitoring systems: A proof-of-concept for yawn detection and seatbelt state detection using an event camera. *IEEE Access*, **2023**.

[255] Michał Żołnowski, Rafal Reszelewski, Diederik Paul Moeys, T Delbrück, and Krzysztof Kamiński. Observational evaluation of event cameras performance in optical space surveillance. In *NEO and Debris Detection Conference, Darmstadt, Germany*. **2019**.

[256] Saeed Afshar, Andrew Peter Nicholson, Andre Van Schaik, and Gregory Cohen. Event-based object detection and tracking for space situational awareness. *IEEE Sensors Journal*, 20(24):15117–15132, **2020**.

[257] Nikolaus Salvatore and Justin Fletcher. Learned event-based visual perception for improved space object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2888–2897. **2022**.

[258] Anastasios N Angelopoulos, Julien NP Martel, Amit PS Kohli, Jorg Conradt, and Gordon Wetzstein. Event based, near eye gaze tracking beyond 10,000 hz. *arXiv preprint arXiv:2004.03577*, **2020**.

[259] Cian Ryan, Brian O'Sullivan, Amr Elrasad, Aisling Cahill, Joe Lemley, Paul Kielty, Christoph Posch, and Etienne Perot. Real-time face & eye tracking and blink detection using event cameras. *Neural Networks*, 141:87–97, **2021**.

[260] Qinyu Chen, Zuowen Wang, Shih-Chii Liu, and Chang Gao. 3et: Efficient event-based eye tracking using a change-based convlstm network. In *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 1–5. IEEE, **2023**.

[261] Ahmed Nabil Belbachir, Martin Litzenberger, Stephan Schraml, Michael Hofstätter, D Bauer, Peter Schön, Martin Humenberger, Christoph Sulzbachner, Tommi Lunden, and M Merne. Care: A dynamic stereo vision sensor system for

fall detection. In *2012 IEEE international symposium on circuits and systems (ISCAS)*, pages 731–734. IEEE, **2012**.

[262] Martin Humenberger, Stephan Schraml, Christoph Sulzbachner, Ahmed Nabil Belbachir, Agoston Srp, and Ferenc Vajda. Embedded fall detection with a neural network and bio-inspired stereo vision. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 60–67. IEEE, **2012**.

[263] Guang Chen, Sanqing Qu, Zhijun Li, Haitao Zhu, Jiaxuan Dong, Min Liu, and Jörg Conradt. Neuromorphic vision-based fall localization in event streams with temporal–spatial attention weighted network. *IEEE Trans. Cybern.*, 52(9):9251–9262, **2022**.

[264] Ganchao Tan, Yang Wang, Han Han, Yang Cao, Feng Wu, and Zheng-Jun Zha. Multi-grained spatio-temporal features perceived network for event-based lip-reading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20094–20103. **2022**.

[265] Hugo Bulzomi, Marcel Schweiker, Amélie Gruel, and Jean Martinet. End-to-end neuromorphic lip-reading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4100–4107. **2023**.

[266] Wenhao Zhang, Jun Wang, Yong Luo, Lei Yu, Wei Yu, and Zheng He. Mtga: Multi-view temporal granularity aligned aggregation for event-based lip-reading. *arXiv preprint arXiv:2404.11979*, **2024**.

[267] Kai Zhang, Yuelei Zhao, Zhiqin Chu, and Yan Zhou. Event-based vision in magneto-optic kerr effect microscopy. *AIP Advances*, 12(9), **2022**.

[268] Clément Cabriel, Tual Monfort, Christian G Specht, and Ignacio Izeddin. Event-based vision sensor for fast and dense single-molecule localization microscopy. *Nature Photonics*, 17(12):1105–1113, **2023**.

[269] Moritz Beck, Georg Maier, Merle Flitter, Robin Gruna, Thomas Längle, Michael Heizmann, and Jürgen Beyerer. An extended modular processing pipeline for event-based vision in automatic visual inspection. *Sensors*, 21(18):6143, **2021**.

[270] Mohammed Salah, Abdulla Ayyad, Mohammed Ramadan, Yusra Abdulrahman, Dewald Swart, Abdelqader Abusafieh, Lakmal Seneviratne, and Yahya Zweiri. High speed neuromorphic vision-based inspection of countersinks in automated manufacturing processes. *Journal of Intelligent Manufacturing*, pages 1–15, **2023**.

[271] Friedhelm Hamann and Guillermo Gallego. Stereo co-capture system for recording and tracking fish with frame-and event cameras. *arXiv preprint arXiv:2207.07332*, **2022**.

[272] Friedhelm Hamann, Suman Ghosh, Ignacio Juarez Martinez, Tom Hart, Alex Kacelnik, and Guillermo Gallego. Low-power, continuous remote behavioral localization with event cameras. *arXiv preprint arXiv:2312.03799*, **2023**.

[273] Guang Chen, Fa Wang, Xiaoding Yuan, Zhijun Li, Zichen Liang, and Alois Knoll. Neurobiometric: An eye blink based biometric authentication system using an event-based neuromorphic vision sensor. *IEEE/CAA Journal of Automatica Sinica*, 8(1):206–218, **2020**.

[274] Annamalai Lakshmi, Anirban Chakraborty, and Chetan S Thakur. Neuromorphic vision: From sensors to event-based algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(4):e1310, **2019**.

[275] Lea Steffen, Daniel Reichard, Jakob Weinland, Jacques Kaiser, Arne Roennau, and Rüdiger Dillmann. Neuromorphic stereo vision: A survey of bio-inspired sensors and algorithms. *Frontiers in neurorobotics*, 13:28, **2019**.

[276] Guang Chen, Hu Cao, Jorg Conradt, Huajin Tang, Florian Rohrbein, and Alois Knoll. Event-based neuromorphic vision for autonomous driving: a paradigm shift for bio-inspired visual sensing and perception. *IEEE Signal Processing Magazine*, 37(4):34–49, **2020**.

[277] Kunping Huang, Sen Zhang, Jing Zhang, and Dacheng Tao. Event-based simultaneous localization and mapping: A comprehensive survey. *arXiv preprint arXiv:2304.09793*, **2023**.

[278] Waseem Shariff, Mehdi Sefidgar Dilmaghani, Paul Kielty, Mohamed Moustafa, Joe Lemley, and Peter Corcoran. Event cameras in automotive sensing: A review. *IEEE Access*, **2024**.

[279] Zexi Jia, Kaichao You, Weihua He, Yang Tian, Yongxiang Feng, Yaoyuan Wang, Xu Jia, Yihang Lou, Jingyi Zhang, Guoqi Li, et al. Event-based semantic segmentation with posterior attention. *IEEE Trans. Image Process.*, **2023**.

[280] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *Int. J. Robot. Res.*, 36(2):142–149, **2017**.

[281] Rui Graça, Brian McReynolds, and Tobi Delbruck. Shining light on the dvs pixel: A tutorial and discussion about biasing and optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4044–4052. **2023**.

[282] Xiao Jiang and Fei Zhou. Charaterization of dim light response in dvs pixel: Discontinuity of event triggering time. *arXiv preprint arXiv:2404.17771*, **2024**.

[283] Ahmed Nabil Belbachir, Stephan Schraml, Manfred Mayerhofer, and Michael Hofstatter. A novel hdr depth camera for real-time 3d 360-degree panoramic vision. In *CVPRW*, pages 419–426. **2014**.

[284] Gottfried Munda, Christian Reinbacher, and Thomas Pock. Real-time intensity-image reconstruction for event cameras using manifold regularisation. *Int. J. Comput. Vis.*, 126(12):1381–1393, **2018**.

[285] Zelin Zhang, Anthony J Yezzi, and Guillermo Gallego. Formulating event-based image reconstruction as a linear inverse problem with deep regularization using optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8372–8389, **2022**.

[286] Souptik Barua, Yoshitaka Miyatani, and Ashok Veeraraghavan. Direct face detection and video reconstruction from event cameras. In *WACV*, pages 1–9. **2016**.

[287] Lin Wang, Tae-Kyun Kim, and Kuk-Jin Yoon. EventSR: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. In *CVPR*, pages 8315–8325. **2020**.

[288] Patrick Alexander Bardow. *Estimating general motion and intensity from event cameras*. Ph.D. thesis, Imperial College London, **2018**.

[289] Timo Stoffregen, Cedric Scheerlinck, Davide Scaramuzza, Tom Drummond, Nick Barnes, Lindsay Kleeman, and Robert Mahony. Reducing the sim-to-real gap for event cameras. In *ECCV*. **2020**.

[290] S. Mohammad Mostafavi, Jonghyun Choi, and Kuk-Jin Yoon. Learning to super resolve intensity images from events. In *CVPR*, pages 2768–2776. **2020**.

[291] Pablo Rodrigo Gantier Cadena, Yeqiang Qian, Chunxiang Wang, and Ming Yang. SPADE-E2VID: Spatially-adaptive denormalization for event-based video reconstruction. *IEEE Trans. Image Process.*, 30:2488–2500, **2021**.

[292] Lin Zhu, Xiao Wang, Yi Chang, Jianing Li, Tiejun Huang, and Yonghong Tian. Event-based video reconstruction via potential-assisted spiking neural network. In *CVPR*, pages 3594–3604. **2022**.

[293] Amit Agrawal, Rama Chellappa, and Ramesh Raskar. An algebraic approach to surface reconstruction from gradient fields. In *ICCV*, volume 1, pages 174–181. **2005**.

[294] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. ESIM: an open event camera simulator. In *CoRL*, pages 969–982. **2018**.

[295] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, pages 2337–2346. **2019**.

[296] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008. **2017**.

[297] Lin Zhu, Jianing Li, Xiao Wang, Tiejun Huang, and Yonghong Tian. Neuspike-net: High speed video reconstruction via bio-inspired neuromorphic cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2400–2409. **2021**.

[298] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11):4311–4322, **2006**.

[299] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 1125–1134. **2017**.

[300] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. **2015**.

[301] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *NeurIPS*, 28:802–810, **2015**.

[302] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, **2014**.

[303] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. *arXiv preprint arXiv:1608.06019*, **2016**.

[304] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pages 2223–2232. **2017**.

[305] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, pages 2462–2470. **2017**.

[306] Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *CVPR*, pages 3867–3876. **2018**.

[307] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI'81: 7th international joint conference on Artificial intelligence*, volume 2, pages 674–679. **1981**.

[308] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883. **2016**.

[309] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, **2021**.

[310] Pengfei Zong, Quanli Liu, He Deng, and Yan Zhuang. Single pixel event tensor: A new representation method of event stream for image reconstruction. *IEEE Sensors Journal*, **2023**.

[311] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, **2020**.

[312] Siying Liu and Pier Luigi Dragotti. Sensing diversity and sparsity models for event generation and video reconstruction from events. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2023**.

[313] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 57(11):1413–1457, **2004**.

[314] Vishal Monga, Yuelong Li, and Yonina C Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44, **2021**.

[315] Qiang Qu, Yiran Shen, Xiaoming Chen, Yuk Ying Chung, and Tongliang Liu. E2hqv: High-quality video generation from event camera via theory-inspired model-aided deep learning. *arXiv preprint arXiv:2401.08117*, **2024**.

[316] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE Trans. Image Process.*, 20(8):2378–2386, **2011**.

[317] Cedric Scheerlinck, Henri Rebecq, Timo Stoffregen, Nick Barnes, Robert Mahony, and Davide Scaramuzza. CED: color event camera dataset. In *CVPRW*. **2019**.

[318] Garrick Orchard, Ajinkya Jayawant, Gregory K Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in neuroscience*, 9:437, **2015**.

[319] Tong Qin, Peiliang Li, and Shaojie Shen. VINS-Mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, **2018**.

[320] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, **2018**.

[321] Zhengqi Li and Noah Snavely. MegaDepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050. **2018**.

[322] Alex Zihao Zhu, Dinesh Thakur, Tolga Özaslan, Bernd Pfrommer, Vijay Kumar, and Kostas Daniilidis. The multivehicle stereo event camera dataset: An event camera dataset for 3d perception. *IEEE Robot. Autom. Lett.*, 3(3):2032–2039, **2018**.

[323] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170. **2017**.

[324] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, **2020**.

[325] Georgios D Evangelidis and Emmanouil Z Psarakis. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE transactions on pattern analysis and machine intelligence*, 30(10):1858–1865, **2008**.

[326] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. DSEC: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 6(3):4947–4954, **2021**.

[327] Stepan Tulyakov, Alfredo Bochicchio, Daniel Gehrig, Stamatios Georgoulis, Yuanyou Li, and Davide Scaramuzza. Time lens++: Event-based frame interpolation with parametric non-linear flow and multi-scale fusion. In *CVPR*, pages 17755–17764. **2022**.

[328] Jeffrey Delmerico, Titus Cieslewski, Henri Rebecq, Matthias Faessler, and Davide Scaramuzza. Are we ready for autonomous drone racing? the uzh-fpv drone racing dataset. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6713–6719. IEEE, **2019**.

[329] B Son, Y Suh, S Kim, H Jung, JS Kim, C Shin, K Park, K Lee, J Park, J Woo, et al. A 640× 480 dynamic vision sensor with a 9 um pixel and 300 meps address-event representation. In *Proceedings of the IEEE International Conference on Solid-State Circuits, San Francisco, CA, USA*, pages 5–9. **2017**.

[330] Ziwei Wang, Liyuan Pan, Yonhon Ng, Zheyu Zhuang, and Robert Mahony. Stereo hybrid event-frame (shef) cameras for 3d perception. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9758–9764. IEEE, **2021**.

[331] Thomas Finateu, Atsumi Niwa, Daniel Matolin, Koya Tsuchimoto, Andrea Mascheroni, Etienne Reynaud, Pooria Mostafalu, Frederick Brady, Ludovic Chotard, Florian LeGoff, et al. 5.10 a 1280× 720 back-illuminated stacked temporal contrast event-based vision sensor with 4.86 μm pixels, 1.066 geps readout, programmable event-rate controller and compressive data-formatting pipeline. In *ISSCC*, pages 112–114. **2020**.

[332] Burak Ercan, Onur Eker, Aykut Erdem, and Erkut Erdem. EVREAL: Towards a comprehensive benchmark and analysis suite for event-based video

reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3942–3951. **2023**.

[333] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 32:8026–8037, **2019**.

[334] Burak Ercan, Onur Eker, Canberk Saglam, Aykut Erdem, and Erkut Erdem. HyperE2VID: Improving event-based video reconstruction via hypernetworks. *IEEE Transactions on Image Processing*, 33:1826–1837, **2024**. doi:10.1109/TIP.2024.3372460.

[335] Chaofeng Chen and Jiadi Mo. IQA-PyTorch: Pytorch toolbox for image quality assessment. [Online]. Available: `https://github.com/chaofengc/IQA-PyTorch`, **2022**.

[336] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. v2e: From video frames to realistic DVS events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1312–1321. **2021**.

[337] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7464–7475. **2023**.

[338] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International journal of computer vision*, 88:303–308, **2009**.

[339] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, **2004**.

[340] Luc Oth, Paul Furgale, Laurent Kneip, and Roland Siegwart. Rolling shutter camera calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1360–1367. **2013**.

[341] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, **2016**.

[342] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2021**.

[343] Ze Wang, Zichen Miao, Jun Hu, and Qiang Qiu. Adaptive convolutions with per-pixel dynamic filter atom. In *ICCV*, pages 12302–12311. **2021**.

[344] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. *NeurIPS*, 29, **2016**.

[345] Yuval Nirkin, Lior Wolf, and Tal Hassner. HyperSeg: Patch-wise hypernetwork for real-time semantic segmentation. In *CVPR*, pages 4061–4070. **2021**.

[346] Tamar Rott Shaham, Michaël Gharbi, Richard Zhang, Eli Shechtman, and Tomer Michaeli. Spatially-adaptive pixelwise networks for fast image translation. In *CVPR*, pages 14882–14891. **2021**.

[347] Brandon Yang, Gabriel Bender, Quoc V Le, and Jiquan Ngiam. Condconv: Conditionally parameterized convolutions for efficient inference. *NeurIPS*, 32, **2019**.

[348] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *CVPR*, pages 11030–11039. **2020**.

[349] Hang Su, Varun Jampani, Deqing Sun, Orazio Gallo, Erik Learned-Miller, and Jan Kautz. Pixel-adaptive convolutional neural networks. In *CVPR*, pages 11166–11175. **2019**.

[350] Jin Chen, Xijun Wang, Zichao Guo, Xiangyu Zhang, and Jian Sun. Dynamic region-aware convolution. In *CVPR*, pages 8064–8073. **2021**.

[351] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, pages 764–773. **2017**.

[352] Yunhao Zou, Yinqiang Zheng, Tsuyoshi Takatani, and Ying Fu. Learning to reconstruct high speed and high dynamic range videos from events. In *CVPR*, pages 2024–2033. **2021**.

[353] Jin Han, Yixin Yang, Chu Zhou, Chao Xu, and Boxin Shi. Evintsr-net: Event guided multiple latent frames reconstruction and super-resolution. In *ICCV*, pages 4882–4891. **2021**.

[354] Patricia Vitoria, Stamatios Georgoulis, Stepan Tulyakov, Alfredo Bochicchio, Julius Erbach, and Yuanyou Li. Event-based image deblurring with dynamic motion awareness. In *ECCVW*, pages 95–112. **2023**.

[355] Bochen Xie, Yongjian Deng, Zhanpeng Shao, Hai Liu, and Youfu Li. Vmv-gcn: Volumetric multi-view based graph cnn for event stream classification. *IEEE Robot. Autom. Lett.*, 7:1976–1983, **2022**.

[356] Yongcheng Jing, Yiding Yang, Xinchao Wang, Mingli Song, and Dacheng Tao. Turning frequency to resolution: Video super-resolution via event cameras. In *CVPR*, pages 7772–7781. **2021**.

[357] Zeyu Xiao, Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Eva^2: Event-assisted video frame interpolation via cross-modal alignment and aggregation. *IEEE Trans. Comput. Imaging*, 8:1145–1158, **2022**.

[358] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456. **2015**.

[359] Ronald J Williams and Jing Peng. An efficient gradient-based algorithm for on-line training of recurrent network trajectories. *Neural computation*, 2(4):490–501, **1990**.

[360] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, **2016**.

[361] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6228–6237. **2018**.

[362] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, pages 41–48. **2009**.

[363] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034. **2015**.

[364] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, **2019**.

[365] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, **2014**.

[366] Lukas Biewald. Experiment tracking with weights and biases, **2020**. Software available from wandb.com.

[367] Cedric Scheerlinck, Henri Rebecq, Timo Stoffregen, Nick Barnes, Robert Mahony, and Davide Scaramuzza. CED: color event camera dataset. In *CVPRW*. **2019**.

[368] Stefan Van der Walt, Johannes L Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. scikit-image: image processing in python. *PeerJ*, 2:e453, **2014**.

[369] Andrew Aitken, Christian Ledig, Lucas Theis, Jose Caballero, Zehan Wang, and Wenzhe Shi. Checkerboard artifact free sub-pixel convolution: A note on sub-pixel convolution, resize convolution and convolution resize. *arXiv preprint arXiv:1707.02937*, **2017**.

[370] Yusuke Sugawara, Sayaka Shiota, and Hitoshi Kiya. Super-resolution using convolutional neural networks without any checkerboard artifacts. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 66–70. IEEE, **2018**.

[371] Jalees Nehvi, Vladislav Golyanik, Franziska Mueller, Hans-Peter Seidel, Mohamed Elgharib, and Christian Theobalt. Differentiable event stream simulator for non-rigid 3d tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1302–1311. **2021**.

[372] Damien Joubert, Alexandre Marcireau, Nic Ralph, Andrew Jolley, André Van Schaik, and Gregory Cohen. Event camera simulator improvements via characterized parameters. *Frontiers in Neuroscience*, 15:702765, **2021**.

[373] Songnan Lin, Ye Ma, Zhenhua Guo, and Bihan Wen. Dvs-voltmeter: Stochastic process-based event simulator for dynamic vision sensors. In *European Conference on Computer Vision*, pages 578–593. Springer, **2022**.

[374] Zhongyang Zhang, Shuyang Cui, Kaidong Chai, Haowen Yu, Subhasis Dasgupta, Upal Mahbub, and Tauhidur Rahman. V2ce: Video to continuous events simulator. *arXiv preprint arXiv:2309.08891*, **2023**.

[375] Pablo Rodrigo Gantier Cadena, Yeqiang Qian, Chunxiang Wang, and Ming Yang. Sparse-e2vid: A sparse convolutional model for event-based video

reconstruction trained with real event noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4149–4157. **2023**.