SNN-ANN Hybrid Networks for Embedded Multimodal Monocular Depth Estimation

Sadia Anjum Tumpa*, Anusha Devulapally*, Matthew Brehove†, Espoir Kyubwa†, Vijaykrishnan Narayanan*
**School of Electrical Engineering and Computer Science*, *Pennsylvania State University*, University Park, PA, USA
†*ChromoLogic LLC*, Monrovia, CA, USA
*sbt5360@psu.edu, *akd5994@psu.edu, †mbrehove@chromologic.com, †ekyubwa@chromologic.com, *vxn9@psu.edu

***Abstract*—Monocular depth estimation is a crucial task in many embedded vision systems with numerous applications in autonomous driving, robotics and augmented reality. Traditional methods often rely only on frame-based approaches, which struggle in dynamic scenes due to their limitations, while event-based cameras offer complementary high temporal resolution, though they lack spatial resolution and context. We propose a novel embedded multimodal monocular depth estimation framework using a hybrid spiking neural network (SNN) and artificial neural network (ANN) architecture. This framework leverages a custom accelerator, TransPIM for efficient transformer deployment, enabling real-time depth estimation on embedded systems. Our approach leverages the advantages of both frame-based and event-based cameras, where SNN extracts low-level features and generates sparse representations from events, which are then fed into an ANN with frame-based input for estimating depth. The SNN-ANN hybrid architecture allows for efficient processing of both RGB and event data showing competitive performance across different accuracy metrics in depth estimation with standard benchmark MVSEC and DENSE dataset. To make it accessible to embedded system we deploy it on TransPIM enabling 9× speedup and 183× lower energy consumption compared to standard GPUs opening up new possibilities for various embedded system applications.**

***Index Terms*—Monocular Depth Estimation, Hybrid Network, Neuromorphic Computing, TransPIM, Neuromorphic Sensor.**

I. Introduction

Depth estimation (DE) is a prevalent task in computer vision that predicts depth from one or more two-dimensional (2D) images with a plethora of applications, including robotics, autonomous driving, 3D image reconstructions, augmented reality, computer graphics, and computational photography. With the progress of recent deep learning (DL) models, DE based on DL models has demonstrated its exceptional efficiency in these wide ranges of applications. Functionally DE can be divided into three divisions [1], including monocular depth estimation (MDE), binocular depth estimation, and multi-view depth estimation amongst which MDE is significantly challenging in computer vision where no reliable cues exist to perceive depth from a sequence of images extracted from a single camera. Nevertheless, this simplicity and ease of access are some of the main advantages of MDE compared to other categories, which require additional complicated pieces of equipment. Because of that, there has been a substantial increase in demand and popularity for MDE in recent years.

Event cameras like Dynamic Vision Sensors (DVS) [2] and Asynchronous Time Based Image Sensor (ATIS) [3] are bio-inspired revolutionary vision sensors only track changes of intensity at the pixel level (referred to as *events*) at the time they occur, asynchronously instead of repeating redundant frame information (i.e., when the camera or the scene is not moving). This creates a stream of events recording the time, location and changes in brightness for that particular location. Event cameras offer key advantages such as low latency, low power, high temporal resolution, high dynamic range, and no motion blur which allow them to operate in extreme conditions (e.g., night, bright sun, rapid motion). In contrast, conventional frame-based cameras capture intensity images at a fixed frame rate providing rich texture and context information. These complementary key aspects of both modalities inspire fusing image frames and event data to improve the overall depth estimation accuracy.

Spiking Neural Networks (SNNs) with their asynchronous event-driven nature of computation show immense potential for extracting the spatio-temporal features from event streams. The core of SNN is the Leaky Integrate and Fire (LIF) neuron which enables SNNs to learn timing information without explicit temporal encoding. However, training deep SNNs is difficult because of vanishing gradients at deeper layers. In this context, we propose a deep hybrid neural network architecture, integrating SNNs and ANNs in different layers, for energy-efficient depth estimation using image frames and event camera data that not only combines the strengths of both data modalities but also exploits the best aspects of SNN and ANN. We are summarizing our contributions below:

- We propose a hybrid SNN-ANN architecture for dense monocular depth estimation fusing image frames and events
- We report competitive performances of our model compared to monocular depth estimation counterparts on the MVSEC and DENSE dataset across different metrics
- We map our compute heavy encoder to a custom accelerator making our model more amenable for embedded system enabling 9× speedup and 183× lower energy consumption compared to NVIDIA TITAN RTX GPU.

This work is supported in part by National Science Foundation (NSF) grant no. 1955815 and DoD N6833522C0487. We would like to acknowledge Minxuan Zhou for providing the TransPIM benchmarking tool. We also thank Chonghan Lee and Rishika Kushwah for their timely help and feedback.

II. Related Works

With the advent of deep learning, especially convolutional neural networks (CNNs), depth estimation has seen significant improvements in accuracy and efficiency. Investigating depth information from RGB images with ground truth labels to train a model using DL in supervised fashion was pioneered in [4] which is a multi-scale CNN-based network consisting of two deep network stacks. Following their work, plethora of other works [5], [6] have been proposed to increase the precision of estimated depth map. Another work presented in [7] proposed to predict both depth and pose simultaneously in a self supervised manner from image frames. However, RGB-based depth estimation struggles with lighting variations, absence of direct depth information and low-texture surfaces.

In [8] learning-based approach was introduced for stereo depth estimation with event cameras yielding dense depth predictions. An unsupervised to estimate depth and optical flow was proposed in [9] that leverages event streams. To estimate monocular dense depth map from events a U-Net architecture is proposed in [10]. While events utilize temporal information to predict depth, they often lack the necessary spatial information, which can limit the accuracy of depth map predictions. Incorporating multiple data modalities to improve the accuracy of depth estimation inspired [11] to propose a recurrent asynchronous network that introduced fusion of event-based data and RGB frames for monocular depth estimation, However, these approaches typically rely on ANNs and do not fully exploit the spatio-temporal information inherent in event data. SNNs have emerged as a promising framework for processing event-based data due to their inherent event-driven nature. An spiking U-Net based SNN has been proposed in [12] to estimate monocular dense depth. While SNNs offer advantages in processing event-based data, it's challenging to train deep SNNs for vanishing gradient in the deeper layers. Given these considerations, a hybrid strategy emerges as an appealing choice for designing deep network structures, harnessing the key advantages of both SNN and ANN. [13] has inspired to explore the hybrid SNN-ANN architecture which proposed a U-Net based optical flow estimation model combining SNNs with traditional ANNs. In this work, we propose a novel hybrid SNN-ANN architecture for monocular depth estimation that leverages the complementary strengths of event-based data and frame-based images aiming at accurate and efficient depth estimation by combining the best aspects of SNNs and ANNs in a unified framework.

III. Hybrid SNN-ANN Training on Multimodal Data

In this section we describe the methodology of our monocular depth estimation workload in details. We have performed the experiments using a widely popular deep learning framework, Pytorch [14] and executed on servers with NVIDIA's A100 80GB PCIe GPUs.

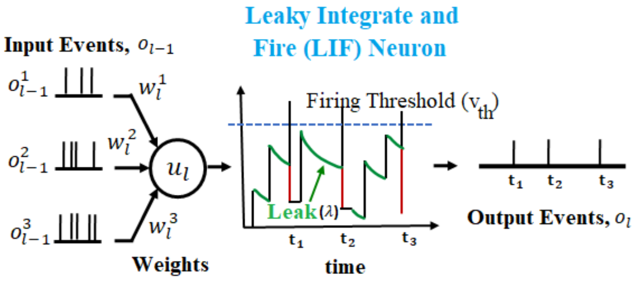

Fig. 1: LIF model of spiking neuron

A. Dataset

We use Multi Vehicle Stereo Event Camera (MVSEC) dataset [15] for training and testing our network. Due to its extensive size and variability, it has emerged as one of the most popular benchmarks for depth construction. MVSEC provides grayscale images and corresponding event streams from two (left and right) synchronized and calibrated Dynamic Vision and Active Pixel Sensors (DAVIS-m346b) from mounted on several vehicles such as a car, motorcycle and hexacopter, with long indoor and outdoor sequences in a variety of illuminations and speeds with a resolution of 346×260 pixels. The ground truth depth map is provided every 50ms by a Velodyne Puck Lite LIDAR mounted on the top of the two event cameras having a sampling frequency of 20 Hz. For estimating monocular depth in our work, we used the grayscale images, events and ground truth provided by the *left* camera. We explored outdoor scenarios by following the same training, validation and test split as [10] and taking a subset of the data. Specifically, the training and validation utilized the largest *Outdoor Day2* sequence, while we tested on *Outdoor Night1*, *Outdoor Night2*, *Outdoor Night3* & *Outdoor Day1* sequences with 1066, 115 and on an average 5000 samples for training, validation and testing respectively.

We also experimented with Depth Estimation oN Synthetic Events (DENSE) dataset [10] which is a synthetic dataset generated from CARLA [16] having five sequences for training (*Towns 01-05*; 5000 samples), two sequences for validation (*Towns 06* & *07*; 2000 samples), and one sequence for testing (*Town 10*; 1000 samples), a total of eight sequences. For our experimentation, the RGB images, the events and the depth maps provided by DENSE dataset are used by the network for training.

B. Data Representation

Event cameras outputs independent pixels responding to the changes in the logarithm of brightness, denoted as $L(\mathbf{u}, t)$ that produces a stream of asynchronous events. When the magnitude of the logarithm of brightness changes by more than a threshold C since the last event, a new event $e_k = (x_k, y_k, t_k, p_k)$ is triggered at the pixel location $\mathbf{u} = (x_k, y_k)^T$ where the event polarity $p_k \in \{+1, -1\}$ denotes the sign of this brightness change. At timestamp t_k an event with polarity p_k is triggered at pixel $\mathbf{u}_k$ when:

$$\Delta L(\mathbf{u}_k, t_k) = p_k(L(\mathbf{u}_k, t_k) - L(\mathbf{u}_k, t_k - \Delta t_k)) \geq C \quad (1)$$

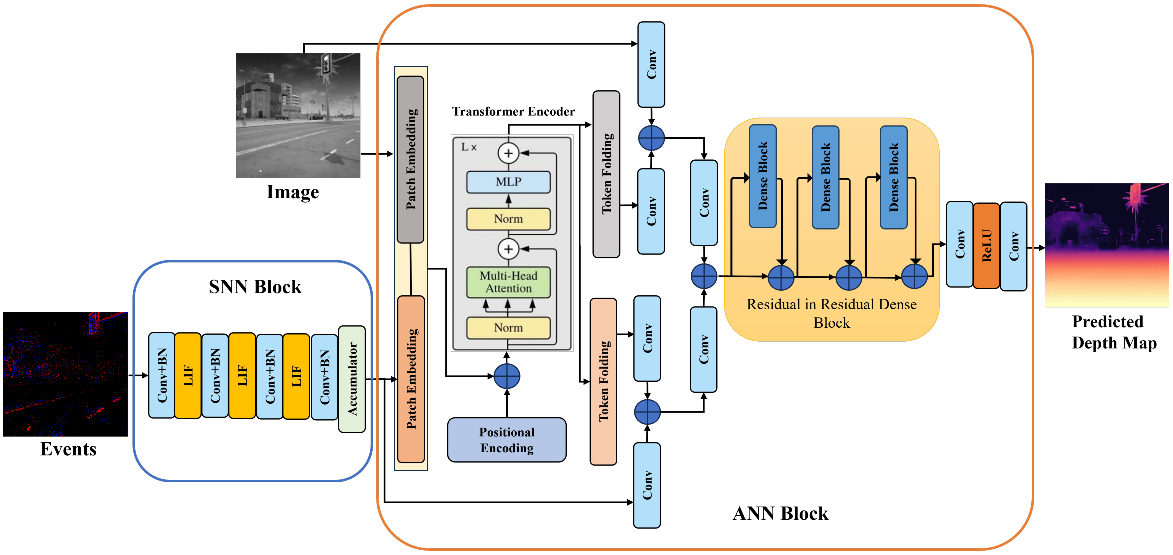

Fig. 2: SNN-ANN Hybrid network architecture

where Δt_k denotes the time since the last event at the same pixel. Each event alone carries very little information of the scene. It is standard practice to aggregate the little information conveyed by individual events into some dense representation before feeding it to the network. Spiking neural networks are well known for their ability to learn temporal information from inputs by utilizing the membrane potentials in neurons We adopted the spatio-temporal voxel-grid representation as presented in [17]. For a set of N input events $\{e_k\}_{k=0}^{N-1}$ between two consecutive grayscale images in the time window $\Delta T = t_{N-1} - t_0$, a set of B event bins are created. The discretized voxel-grids are generated using bilinear sampling with spatial dimentions $H \times W$ and B temporal bins. In order to facilitate the learning of temporal information by SNNs and hybrid networks from the inputs, we pass these B bins in order. In our experiments, we used $\Delta T = 50ms$ of events and B = 5 temporal bins as in [10].

MVSEC dataset provides 1 channel grayscale images and DENSE dataset provides 3 channel RGB images as image frames. We used data augmentations like random horizontal flipping and random cropping on input frames, event voxels and ground truth depth maps for generalizing our model. Additionally, data cropping to 256×256 resolution before feeding to model ensures compatibility with transformer-based encoder, which expects a square input size. The ground truth depth maps are transformed into normalized log depth maps to better represent depth variations in smaller range.

C. Spiking Neuron Model

The basic unit of biologically inspired SNNs is spiking neuron. In this work we consider the popular leaky-integrate-and-fire (LIF) neuron model [18] for its simplicity and scalability. Once the accumulated input information over time (referred to as *internal state*) of a spiking neuron reaches a pre-defined threshold, it *fires* an output spike and *resets* its state to a resting potential. For digital simulations, a discrete-time formulation of the LIF neuron is as follows:

$$U_t^l = \lambda U_{t-1}^l + W^l o_t^{l-1} - \theta o_{t-1}^l, o_{t-1}^l = \begin{cases} 1, & \text{if } U_{t-1}^l > \theta \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where U_t^l is the neuron's membrane potential (internal neuron state) at time t, l is the layer index, λ (≤ 1) is the membrane potential leak, W_l is the weight matrix connecting layers $l-1$ and l, o_t is the spike vector at time t, and θ is the firing threshold potential. The first term on the right hand side of Eq. 2 carries forward the neuron state from the previous time-step to the current time-step (modulated by leak λ); the second term is a weighted sum of the spikes coming from the previous layer and the third term arises from the thresholding non-linearity that decreases the membrane potential by θ if an output spike o_{t-1}^l is generated by a neuron (second part of Eq. 2). The spike generation mechanism presented in Eq. 2 is shown in Fig. 1

D. Hybrid Network Architecture

The proposed hybrid SNN-ANN architecture represents a class of neural network design that integrates both Spiking Neural Networks and Artificial Neural Networks across different layers. ANNs excel at processing dense data representations, but they struggle with the sparse and asynchronous nature of data from event cameras. On the other hand, while SNNs are well-suited for handling event-based data due to their temporal processing capabilities, they face challenges with training, including issues with vanishing gradients. The motivation behind this integration is to harness the unique strength of each network types for monocular depth estimation.

The architecture of hybrid SNN-ANN is shown in Fig 2. For efficient spatio-temporal processing of event data, we place SNN layers immediately after the input layer as SNNs have inherent capacity to extract event data more effectively. We used 3 convolutional-SNN followed by LIF activations. The output at the 3rd layer of SNN block is passed through a convolutional block and accumulated over all temporal bins

TABLE I: Absolute mean depth errors (in meters) at different cut-off depth distances for MVSEC data. **E** refers to event-based input, **I** refers to image frame-based input and **E+I** represents multimodal fusion of events and image frame based input

Data	**Cut-off Distance**	**MonoDepth [5]**	**MegaDepth [6]**	**Zhu. et al [9]**	**E2Depth [10]**	**StereoSpike [12]**	**RAM Net [11]**	***Hybrid (ours)***	
		I	**I**	**E**	**E**	**E**	**E+I**	**E**	**E+I**
Outdoor Night 1	10m	3.49	2.54	3.13	3.38	1.68	2.50	1.76	**1.18**
	20m	6.33	4.15	4.02	3.82	2.61	3.19	2.66	**2.02**
	30m	9.31	5.60	4.89	4.46	3.18	3.82	**3.04**	3.31
Outdoor Night 2	10m	5.15	3.92	2.19	1.67	Not Reported	**1.21**	2.11	1.24
	20m	7.80	5.78	3.15	2.63		2.31	3.30	**2.18**
	30m	10.03	7.05	3.92	3.58		**3.28**	3.72	3.44
Outdoor Night 3	10m	4.67	4.15	2.86	1.42	Not Reported	**1.01**	1.86	1.15
	20m	8.96	6.00	4.46	2.33		2.34	3.12	**2.03**
	30m	13.36	7.24	5.05	**3.18**		3.43	3.53	3.32
Outdoor Day 1	10m	3.44	2.37	2.72	1.85	1.35	1.39	**1.26**	1.49
	20m	7.02	4.06	3.84	2.64	2.30	2.17	**2.05**	2.41
	30m	10.03	5.38	4.40	3.13	2.75	2.76	**2.41**	2.80

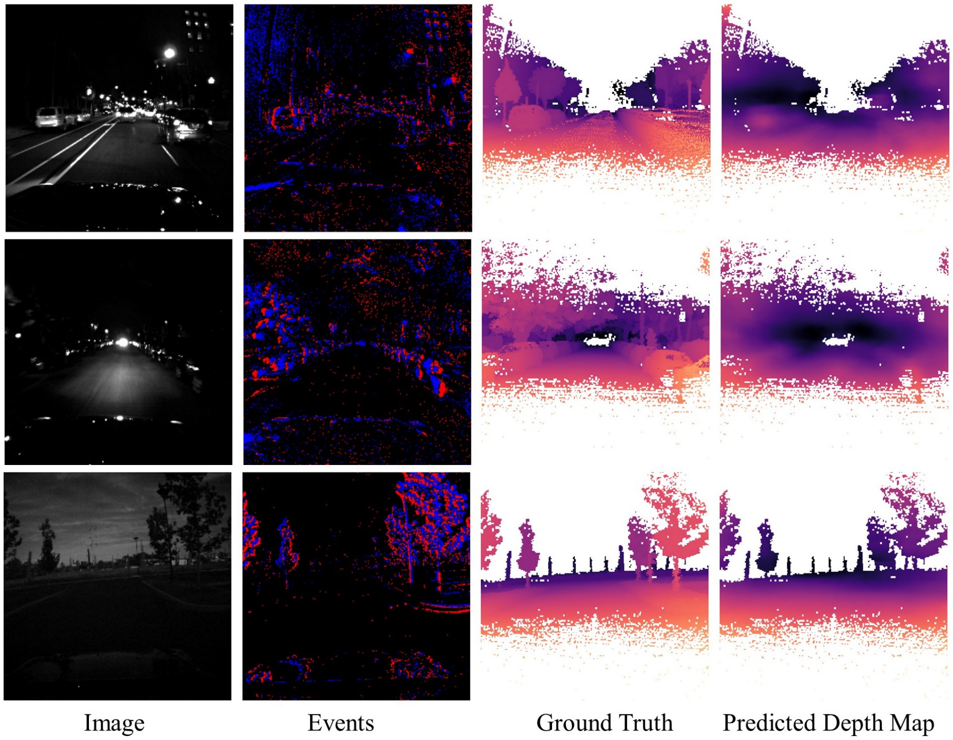

Fig. 3: Qualitative comparison on test sequence outdoor night 1, night 3 and day 1 (row-wise) of MVSEC dataset. Image frame, events, ground truth and predicted depth maps at *valid* pixels are shown respectively from left to right.

and then processed through patch embedding. Image frames also go through patch embedding. As shown in Fig 2, concatenated patches from images and SNN block accumulator are then processed through positional encoding which is the starting point of the ANN block of our hybrid model. Then the tokens are fed to vision transformer (ViT) [19] encoder followed by token folding operation.

Using token folding, the transformer encoder generates a series of tokens that are subsequently reshaped to their original dimension which then undergoes processing in a multimodal fusion block [20] utilizing convolution and skip connections. This late fusion block restores information which ensures the recovery of any potentially lost information. Another set of convolution operation is performed to combine the fused output from both the image and event modalities. This fused output is then passed to the residual in residual dense block (RRDB) [21] that combines multi-level residual network and dense connections fully utilizing hierarchical features which are fed to final convolutional layers to generate depth map with better restoration quality.

E. Training Details

We train our hybrid model from scratch in a supervised manner using the ground truth depth maps. To compute the loss between the *valid* ground truth labels and the prediction outputs, we employ a combination of L_1 loss, normal loss [20], and multi-scale gradient matching loss [6] like this:

$$L_{total} = \alpha L_{l1_{loss}} + \beta L_{normal} + \gamma L_{grad} \tag{3}$$

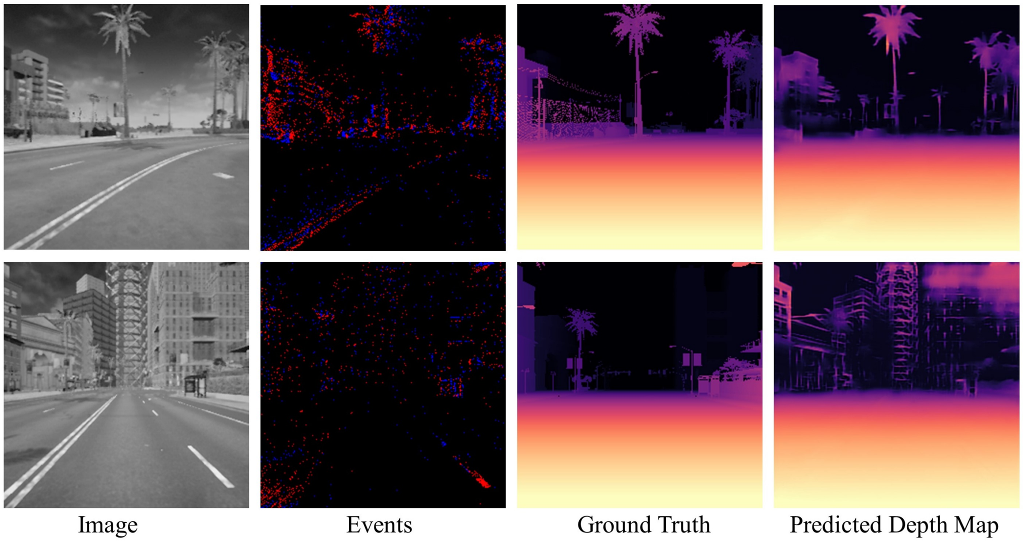

Fig. 4: Qualitative comparison on test sequence town 10 of DENSE dataset. Image frame, events, ground truth and predicted depth maps are shown respectively from left to right.

TABLE II: Quantitative results on the DENSE dataset. Town06 & Town07 for validation and Town10 sequence for testing.

Model	**Dataset**	**Abs Rel↓**	**Sq Rel↓**	**RMSE↓**	**RMSE log↓**	**SI log↓**	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$	**Avg. Error↓ 10m**	**Avg. Error↓ 20m**	**Avg. Error↓ 30m**
E2Depth [10]	Town06	0.120	0.083	**6.640**	0.188	0.035	0.855	0.956	**0.987**	0.31	**0.74**	**1.32**
	Town07	0.267	0.535	10.182	0.328	0.098	0.774	0.878	0.927	1.03	2.35	3.06
	Town10	0.220	0.279	**11.812**	0.323	0.093	0.724	0.865	**0.932**	0.61	1.45	**2.42**
Hybrid (E+I)	Town06	**0.080**	**0.071**	6.987	**0.164**	**0.023**	**0.934**	**0.967**	0.981	**0.16**	0.75	1.44
	Town07	**0.099**	**0.055**	**8.852**	**0.225**	**0.040**	**0.865**	**0.923**	**0.963**	**0.29**	**0.75**	**1.20**
	Town10	**0.176**	**0.239**	13.624	**0.308**	**0.079**	**0.821**	**0.886**	0.929	**0.26**	**1.13**	2.93

TABLE III: Performance Analysis of the Transformer Module.

Components	**Time Taken (ms)**	**Energy Consumption (mJ)**
Trans-PIM [22]	171.08	114.49
NVIDIA TITAN RTX	1412.1	20886.3

While experimenting with MVSEC, the weight coefficient for L_1 loss (α) and weight coefficient for multi-scale gradient matching loss (γ) are 0.5 and 0.25, respectively. We employed a transformer encoder with depth 12, the ADAM optimizer with a batch size of 8, learning rate of 3×10^{-4} and trained the model for 70 epochs. Also, we experimented with different spiking thresholds of LIF neurons and observed that the model is learning well with threshold 1.0. For DENSE dataset, we used the weight coefficient for normal loss (β) and weight coefficient for multi-scale gradient matching loss (γ) are 0.5 and 0.25, respectively and reduced transformer depth of 6 is used with batch size 16.

IV. Evaluation

In this section we will show both qualitative and quantitative results on MVSEC and DENSE dataset.

A. Evaluation Metrics

We evaluated the results using different metrics [4]. Absolute mean depth error [6] at 10m, 20m & 30m cutoff distances measures the absolute difference between the predicted depth and the ground truth depth at different cutoff distances. Absolute relative difference provides a percentage error relative to the ground truth depth. Square relative difference uses squared differences, which penalize larger errors more. RMSE calculates the square root of the average of squared differences between predicted and ground truth depths. RMSE Log and SI Log are suitable for evaluating errors across a wide range of depths. Depth threshold evaluates the percentage of pixels where the predicted depth is within a certain threshold of ground truth depth.

B. Results on MVSEC dataset

The experimental results on real MVSEC are tabulated in Table I. Here the absolute mean depth errors (in meters) at 10m, 20m & 30m cutoff distances are compared with the state-of-the-art monocular depth estimation models for four test datasets of night and day conditions. It is observed that our hybrid model shows competitive performance across all the test datasets. For outdoor day 1, our event-based hybrid model outperforms the SOTA models potentially because event-based model is more robust to dynamic movements and external noise present in day environment. For Night condition there is comparatively less movement in the scene, so, our multimodal hybrid model gives competitive results. Figure 3 illustrates a visual comparison of our hybrid model's depth prediction at *valid* pixels with the MVSEC grayscale image, events input and ground truth depth map.

C. Results on DENSE dataset

The quantitative results on synthetic DENSE dataset are tabulated in Table II comparing our model with [10] for two validation datasets: *Town06 & 07* and for test dataset *Town10*. It is observed that our hybrid model mostly outperforms across all the datasets. Figure 3 illustrates a visual comparison of our hybrid model's depth prediction at each pixel with the DENSE RGB image, events input and ground truth depth map.

D. Using Custom Hardware to Support Embedded System

To evaluate the feasibility of low-latency Vision Transformers (ViTs) in multi-modal fusion for embedded systems, we focus on the computationally expensive encoder block. While ViTs offer strong performance, their deployment is often limited by resource constraints. We address this by employing the current state-of-the-art processing in memory transformer accelerator, TransPIM [22] which is a custom hardware accelerator designed for efficient execution of Transformer models. TransPIM leverages a software-hardware co-design and a hybrid in-memory/near-memory computing paradigm for exceptional performance on emerging High Bandwidth Memory (HBM) architectures. Notably, TransPIM utilizes an 8-bit quantized input format, whereas the standard approach on GPUs is floating-point precision. However, as demonstrated in [23], this difference in data format does not significantly impact the model's accuracy. Our evaluation compares the performance of the ViT encoder block on TransPIM against a standard NVIDIA TITAN RTX. This comparison demonstrates that TransPIM achieves a significant 9× speedup and a substantial 183× reduction in energy consumption compared to the GPU as shown in Table III. This evaluation highlights the potential of TransPIM for enabling low-latency ViT deployments in resource-constrained multi-modal fusion applications.

V. Conclusion and Future Work

We explored a hybrid SNN-ANN model to leverage the advantages of both paradigm to estimate monocular depth with multimodal fusion of event and frame based data. The findings from the experiments show that our proposed hybrid architecture can effectively estimate monocular depth with competitive performance enabled from better temporal information extraction by SNN followed by deep ANN backbone. Moreover, it offers significant improvements with respect to execution speed and energy consumption as observed when the transformer encoder is simulated with a custom hardware accelerator, TransPIM for embedded systems. With the flourishment of neuromorphic chips, as our future work we hope to deploy the SNN part of the network on Loihi which would optimize the model further.

References

[1] A. Masoumian, H. A. Rashwan, J. Cristiano, M. S. Asif, and D. Puig, "Monocular depth estimation using deep learning: A review," *Sensors*, vol. 22, no. 14, 2022.

[2] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[3] C. Posch, D. Matolin, and R. Wohlgenannt, "A qvga 143db dynamic range asynchronous address-event pwm dynamic image sensor with lossless pixel-level video compression," in *IEEE ISSCC*, 2010, pp. 400–401.

[4] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Advances in neural information processing systems*, vol. 27, 2014.

[5] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *IEEE CVPR*, July 2017.

[6] Z. Li and N. Snavely, "Megadepth: Learning single-view depth prediction from internet photos," in *IEEE/CVF CVPR*, 2018, pp. 2041–2050.

[7] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *IEEE CVPR)*, 2017, pp. 6612–6619.

[8] S. Tulyakov, F. Fleuret, M. Kiefel, P. Gehler, and M. Hirsch, "Learning an event sequence embedding for dense event-based deep stereo," in *IEEE/CVF ICCV)*, 2019, pp. 1527–1537.

[9] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Unsupervised event-based learning of optical flow, depth, and egomotion," in *IEEE/CVF CVPR*, 2019, pp. 989–997.

[10] J. Hidalgo-Carri'o, D. Gehrig, and D. Scaramuzza, "Learning monocular dense depth from events," *3DV*, pp. 534–542, 2020.

[11] D. Gehrig, M. Rüegg, M. Gehrig, J. Hidalgo-Carrió, and D. Scaramuzza, "Combining events and frames using recurrent asynchronous multimodal networks for monocular depth prediction," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2822–2829, 2021.

[12] U. Rançon, J. Cuadrado-Anibarro, B. R. Cottereau, and T. Masquelier, "Stereospike: Depth learning with a spiking neural network," *IEEE Access*, vol. 10, pp. 127 428–127 439, 2022.

[13] C. Lee, A. Kosta, A. Z. Zhu, K. Chaney, K. Daniilidis, and K. Roy, "Spike-flownet: Event-based optical flow estimation with energy-efficient hybrid neural networks," in *ECCV*. Springer, 2020, pp. 366–382.

[14] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035.

[15] A. Z. Zhu, D. Thakur, T. Özaslan, B. Pfrommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event camera dataset for 3d perception," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2032–2039, 2018.

[16] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., vol. 78. PMLR, 13–15 Nov 2017, pp. 1–16.

[17] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Unsupervised event-based optical flow using motion compensation," in *ECCV Workshops*, 2018.

[18] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *ArXiv*, vol. abs/2010.11929, 2020.

[20] M. F. F. Khan, A. Devulapally, S. Advani, and V. Narayanan, "Robust multimodal depth estimation using transformer based generative adversarial networks," in *ACM MM*, 2022, p. 3559–3568.

[21] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *ECCV Workshops*. Springer-Verlag, 2019, p. 63–79.

[22] M. Zhou, W. Xu, J. Kang, and T. Rosing, "Transpim: A memory-based acceleration via software-hardware co-design for transformer," in *IEEE HPCA*, 2022, pp. 1071–1085.

[23] Y. Pan, M. Zhou, C. Lee, Z. Li, R. Kushwah, V. Narayanan, and T. Rosing, "Primate: Processing in memory acceleration for dynamic token-pruning transformers," in *ASP-DAC*, 2024, pp. 557–563.