ISSN 2464-4617 (print)
ISSN 2464-4625 (online)
Computer Science Research Notes - CSRN
http://www.wscg.eu
WSCG 2025 Proceedings

6-DOF Pose Estimation For Event Cameras Using A Transformer-Based Approach

Ahmed Tabia
Univ Evry
France 91000

Fabien Bonardi
Univ Evry
France 91000

Samia Bouchaa
Univ Evry
France 91000

ahmed.tabia@universite-paris-saclay.fr

ABSTRACT

Event cameras are novel sensors that provide significant advantages over traditional cameras, such as low latency, high dynamic range, and reduced motion blur. These properties make them particularly well-suited for 6-DOF pose estimation tasks in challenging environments. In this paper, we present a novel transformer-based approach for 6-DOF pose estimation using event camera data. Our method combines a pretrained ResNet50 backbone for feature extraction with a custom transformer encoder to model the spatial and temporal dependencies inherent in event data. We demonstrate the effectiveness of our approach on a dataset of real-world event camera images, where we achieve significant improvements in pose estimation accuracy compared to state-of-the-art methods. Additionally, our method exhibits robustness to varying lighting conditions, motion blur, and sensor noise, highlighting its potential for deployment in a wide range of applications, such as robotics, autonomous vehicles, and augmented reality. Our experimental results showcase the promising capabilities of transformer-based models in leveraging the unique properties of event cameras for accurate and efficient 6-DOF pose estimation.

Keywords

6Dof pose estimation, Deep Learning, Transforms, Event Based Camera.

1 INTRODUCTION

Camera pose relocalization, the task of determining the position and orientation of a camera from an observed scene [27], is a fundamental issue in numerous computer vision applications. These applications range from autonomous vehicle driving and robotics, to augmented reality and pedestrian visual positioning. Conventional relocalization methodologies in computer vision can be grouped into two primary categories: (1) geometric-based and (2) learning-based approaches. Geometric-based strategies [24] predominantly rely on local feature matching. These methods involve extracting local features from an image, performing a 2D-3D match with corresponding 3D points, and subsequently calculating the six-degree-of-freedom camera pose using Perspective-n-Point algorithms [14]. These techniques, however, are heavily dependent on the accuracy of the feature extraction and matching processes, which can be particularly compromised in conditions of variable illumination [15]. In contrast, the emergence of deep learning has brought a renewed focus on learning-based approaches. Traditionally dominated by Convolutional Neural Networks (CNN), learning-based methods have been successful in various tasks such as object recognition [6], image classification [19], and segmentation [1]. These methods have demonstrated robust feature extraction abilities [12] but require large amounts of training data and substantial computational resources. Thus, methods that alleviate the necessity for recomputation, such as transfer learning and integration, can be more compelling. Despite these advancements, both geometric and learning-based relocalization methods continue to struggle with factors like illumination changes, blur, and flat images that make feature extraction challenging. This difficulty is primarily due to the nature of the input images captured by conventional cameras and used in these methods. An alternative approach is using event cameras, also known as neuromorphic cameras. These sensors respond to local brightness changes and produce a stream of asynchronous events (discrete pixel-wise brightness changes) corresponding to scene illumination changes. These cameras offer several advantages over conventional cameras, including high temporal resolution, wide dynamic range, and the absence of motion blur. These benefits make event cameras ideally suited for pose estimation in robotic applications.

Building on these advantages, Rebecq et al.[28] introduced a method combining IMU and event information to estimate the six-degree-of-freedom (6DOF) camera pose. More recently, a method known as SP-LSTM[25] was developed, employing a VGG16 architecture [31] trained from scratch with a stochastic gradient descent algorithm and two stacked spatial LSTM layers. While this method achieved promising results, it required extensive training time due to the retraining of the entire network model with the LSTM layer.

In this paper, we introduce a new method to alleviate the issues associated with LSTM layers [33] [25]. Our approach involves a deep learning model that uses transformers instead of CNNs for feature extraction from event images. We employ a pretrained ResNet50 backbone for feature extraction combined with a custom transformer encoder to model the spatial and temporal dependencies inherent in event data. Following feature extraction, these features are aggregated using the outer product at each image location and pooled using a bilinear pooling operation [17]. We also incorporate advancements in deep learning, employing the ADAM optimizer [13] along with the ELU activation function [2]. Through conducting experiments across multiple datasets, we demonstrate that our method outperforms state-of-the-art methods, confirming its effectiveness and applicability

2 RELATED WORK

In this section, we delve into the predominant techniques in the field, beginning with an examination of pose estimation methodologies for conventional cameras, and then moving on to scrutinize the relatively sparse range of works addressing pose estimation for event cameras.

2.1 Standard methods for pose estimation

The majority of advanced methods for camera pose estimation have been formulated in the context of RGB cameras. These can be broadly divided into two categories: Geometry-based and Learning-based approaches. Geometry-based approaches typically operate on the assumption of a pre-existing three-dimensional environment. Using a set of scene images, these methodologies [16] often construct a three-dimensional model employing principles of Structure from Motion (SfM)[30][9] or Simultaneous Localization and Mapping (SLAM) [23]. A commonality among these geometric-based approaches is the pivotal role of feature extraction and matching processes. On the other hand, learning-based methods are primarily data-driven and rely heavily on learning algorithms. The resurgence of the deep learning paradigm has played a significant role in bolstering the popularity of this category. With the widespread availability of data and rapid progress in computational resources, many researchers are revisiting traditional computer vision applications using these learning-based techniques. This paradigm shift has heralded significant advancements in numerous tasks, with many successful works emerging as a result. Notable examples include the impressive results attained in image classification by models such as ResNet [10] and Inception [32], superior performance in image segmentation by Long et al.[18], and substantial advancements in object detection by Girshick[8] among others. Within the sphere of camera pose estimation, PoseNet, a method introduced by Kendall et al. [12], aims to estimate camera pose from an RGB image. Despite adhering to a deep learning paradigm, the accuracy of PoseNet remains relatively lower compared to traditional methods.

2.2 Event camera pose estimation

Event cameras, known for their low latency, are particularly adept at facilitating real-time motion analysis and high-speed robotics applications. Each pixel is processed independently, thus negating the need for a global exposure time frame [22]. Furthermore, the High Dynamic Range (HDR) ($>$120dB) ensures that these cameras do not succumb to motion blur [29]. Early work with event cameras leveraged these attributes for applications such as swift object tracking to control basic robotic systems [3], steering angle prediction for self-driving cars [20], and 1-megapixel object detection [26]. Authors of [21] implemented an on-board perception system with an event camera for 6DOF pose tracking of a quadrotor, allowing for pose estimation during high-speed maneuvers against a known model. More recently, the method introduced in [7] provided a direct estimation of the camera's angular velocity. Rebecq et al [28] presented a method to merge IMU data with events for 6DOF camera relocalization. A more recent deep learning-based method, known as (SP-LSTM), proposed by[25] estimated 6DOF using a custom architecture. This architecture combined a pretrained ResNet50 backbone for feature extraction with a custom transformer encoder to model the spatial and temporal dependencies inherent in event data. Despite achieving promising results for camera pose estimation, this method required a significant investment in training time due to the retraining of the complete network model. In our work, we build upon these developments and propose to evaluate a range of CNN architectures employed for extracting pertinent pose features. We also suggest a single-layer spatial LSTM, which consolidates and condenses the extracted features. Capitalizing on recent strides made in deep learning, we implement a NADAM optimizer [5] in tandem with the ELU activation function [2]. We execute experiments on real

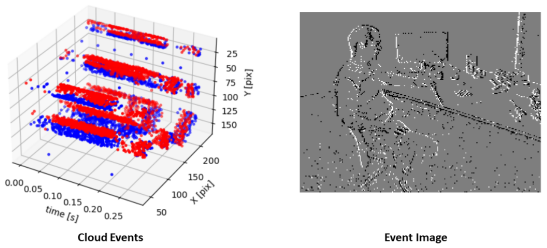

Figure 1: Image preprocessing from cloud of point to event images inspired by [7]

and simulated datasets and present results from various network architectures.

3 PROPOSED METHOD

Our proposed method for 6-DOF pose estimation using event camera data involves a pipeline that consists of two primary steps: feature extraction with ResNet50 and temporal dependency modeling with a Transformer encoder. The details are described below.

3.1 Feature Extraction with ResNet50

In order to convert raw event data into a suitable representation for pose estimation, we leverage the feature extraction capabilities of a ResNet50 [10] convolutional neural network (CNN) pre-trained on ImageNet [4].

The input to this network is an event image, which is generated from a sequence of events captured by the event camera. Given an event sequence, we follow a similar approach to Nguyen et al. [25] to convert these sequences into event images $I \in \mathbb{R}^{H \times W}$, where H and W represent the height and width of the image respectively. Each event e is a tuple, represented as:

$$e = \langle e_t, (e_x, e_y) e_p \rangle, \tag{1}$$

where e_t is the timestamp of the event, (e_x, e_y) are the pixel coordinates, and $e_p = \pm 1$ is the polarity indicating the direction of brightness change at the corresponding pixel.

These event images are then passed through the ResNet50 network to generate a high-dimensional feature vector $F \in \mathbb{R}^D$, where D is the dimension of the feature space.

3.2 Temporal Dependency Modeling with Transformer Encoder

Traditional recurrent models like LSTMs or GRUs are commonly used to model temporal dependencies in sequence data. However, these models process the sequence data in a step-by-step manner, making it challenging to capture long-range dependencies. On the other hand, the Transformer model, introduced by Vaswani et al. [34], can capture such dependencies more efficiently, without the need for sequential processing.

We pass the feature vector F obtained from ResNet50 to a Transformer encoder. The Transformer encoder generates a context-aware representation $T \in \mathbb{R}^D$ for each event image, where each feature not only captures the information of the corresponding event but also considers the information from other events in the sequence.

3.3 Pose Estimation

The final step of our pipeline is to estimate the 6-DOF camera pose from the Transformer outputs. To achieve this, we use a fully connected (FC) layer as a regressor.

This FC layer maps the context-aware feature representation T to a 6-DOF camera pose vector $P \in \mathbb{R}^7$. This mapping can be represented as:

$$P = FC(T), \tag{2}$$

where $FC(.)$ denotes the function represented by the FC layer. The pose vector P consists of three translation components t_x, t_y, t_z and three rotation components represented in Euler angles $\theta_x, \theta_y, \theta_z$.

This completes our proposed method for 6-DOF pose estimation using event camera data. In the following sections, we present experimental evaluations of our method on a real-world event camera dataset.

4 EXPERIMENTAL SETUP

4.1 Dataset

We validate the effectiveness of our method on the Event Camera Dataset [22], which is widely used for evaluating event-based vision algorithms. The dataset comprises sequences captured using a DVS128 event camera in various indoor and outdoor environments. Each sequence in the dataset includes a stream of asynchronous events, accompanied by ground truth 6-DOF camera poses obtained from a high-accuracy motion-capture system.

4.2 Training Details

Our model consists of a ResNet50 backbone, a transformer encoder, and a fully connected layer. The ResNet50 model is initialized with weights pre-trained on ImageNet [4], while the transformer and fully connected layers are randomly initialized. The model is trained using the Adam optimizer [13], with an initial learning rate of $1e-4$. The learning rate is reduced by a factor of 0.1 whenever the validation loss plateaus. The model is trained for 100 epochs with a batch size of 32. We use the Mean Squared Error (MSE) loss between the predicted poses and ground truth poses as

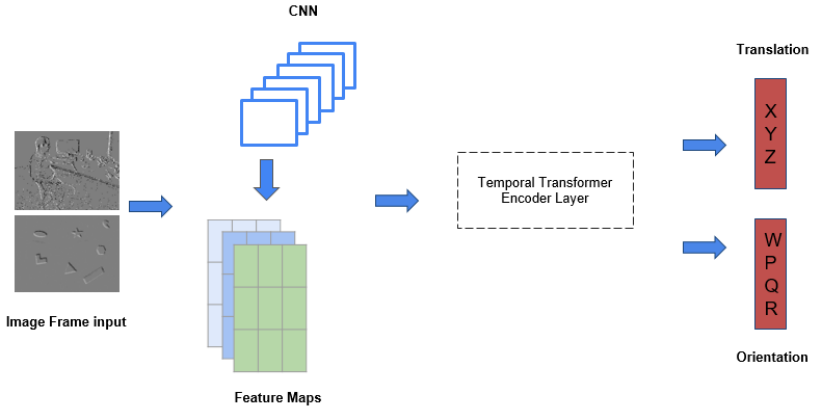

Figure 2: An overview of our 6DOF pose relocalization method for event cameras . We first create an event image from stream of events. Then we extract features from the created event image.Once these event images are generated, they are fed into our feature extraction model, which is built on a ResNet50 architecture pre-trained on ImageNet. This model extracts high-level spatial features from the event images, providing a robust representation of the input data. Then the output is passed to a custom transformer encoder.

	PoseNet [12]		**Bayesian PoseNet** [11]		**SP-LSTM** [25]		**Ours**	
	Median Error	Average Error	Median Error	Average Error	Median Error	Average Error	Median Error	Average Error
shapes rotation	0.109m, 7.388°	0.137m, 8.812°	0.142m, 9.557°	0.164m, 11.312°	0.025m, 2.256°	0.028m, 2.946°	0.020m, 1.893°	0.025m, 2.591°
shapes translation	0.238m, 6.001°	0.252m, 7.519°	0.264m, 6.235°	0.269m, 7.585°	0.035m, 2.117°	0.039m, 2.809°	0.035m, 2.321°	0.032m, 2.523°
box translation	0.193m, 6.977°	0.212m, 8.184°	0.190m, 6.636°	0.213m, 7.995°	0.036m, 2.195°	0.042m, 2.486°	0.032m, 1.977°	0.041m, 1.825°
dynamic 6dof	0.297m, 9.332°	0.298m, 11.242°	0.296m, 8.963°	0.293m, 11.069°	0.031m, 2.047°	0.036m, 2.576°	0.031m, 1.912°	0.042m, 2.594°
hdr poster	0.282m, 8.513°	0.296m, 10.919°	0.290m, 8.710°	0.308m, 11.293°	0.051m, 3.354°	0.060m, 4.220°	0.042m, 2.945°	0.059m, 3.883°
poster translation	0.266m, 6.516°	0.282m, 8.066°	0.264m, 5.459°	0.274m, 7.232°	0.036m, 2.074°	0.041m, 2.564°	0.038m, 2.145°	0.040m, 2.415°
Average	0.231m, 7.455°	0.246m, 9.124°	0.241m, 7.593°	0.254m, 9.414°	0.036m, 2.341°	0.041m, 2.934°	**0,031m, 2.198°**	**0.039m, 2.638°**

Table 1: Comparison between our method results and the results of PoseNet [12], Bayesian PoseNet [11] and SP-LSTM [25]. The evaluation is performed using the random split protocol.

	PoseNet [12]		**Bayesian PoseNet** [11]		**SP-LSTM** [25]		**Ours**	
	Median Error	Average Error	Median Error	Average Error	Median Error	Average Error	Median Error	Average Error
shapes rotation	0.201m, 12.499°	0.214m, 13.993°	0.164m, 12.188°	0.191m, 14.213°	0.045m, 5.017°	0.049m, 11.414°	0.049, 3.581°	0.048m, 6.901°
shapes translation	0.198m, 6.969°	0.222m, 8.866°	0.213m, 7.441°	0.228m, 10.142°	0.072m, 4.496°	0.081m, 5.336°	0.064m, 4.685°	0.071m, 5.384°
shapes 6dof	0.320m, 13.733°	0.330m, 18.801°	0.326m, 13.296°	0.329m, 18.594°	0.078m, 5.524°	0.095m, 9.532°	0.072m, 5.875°	0.093m, 7.652°
Average	0.240m, 11.067°	0.255m, 13.887°	0.234m, 10.975°	0.249m, 14.316°	0.065m, 5.012°	0.075m, 8.761°	**0.061m, 4.713°**	**0.070m, 6.645°**

Table 2: Comparison between our method results and the results of PoseNet [12], Bayesian PoseNet [11] and SP-LSTM [25]. The evaluation is performed using the novel split protocol.

our optimization objective. The MSE loss is defined as follows:

$$L = \frac{1}{N}\sum_{i=1}^{N}(P_i - P_i^{gt})^2, \tag{3}$$

where N is the total number of samples in the dataset, P_i is the predicted pose, and P_i^{gt} is the ground truth pose.

5 RESULTS AND DISCUSSION

The performance of our model is evaluated in terms of translation and rotation errors. The translation error is computed as the Euclidean distance between the predicted and ground truth translations, while the rotation error is calculated as the angle between the predicted and ground truth rotations, converted to Euler angles. Our model achieves competitive results when compared with state-of-the-art methods on the Event Camera Dataset, demonstrating its effectiveness in estimating 6-DOF camera poses from event data.

5.0.1 Comparison with state-of-the-art methods

we report the comparison results between our method explained on the section 3.1 and the state of the art mod-

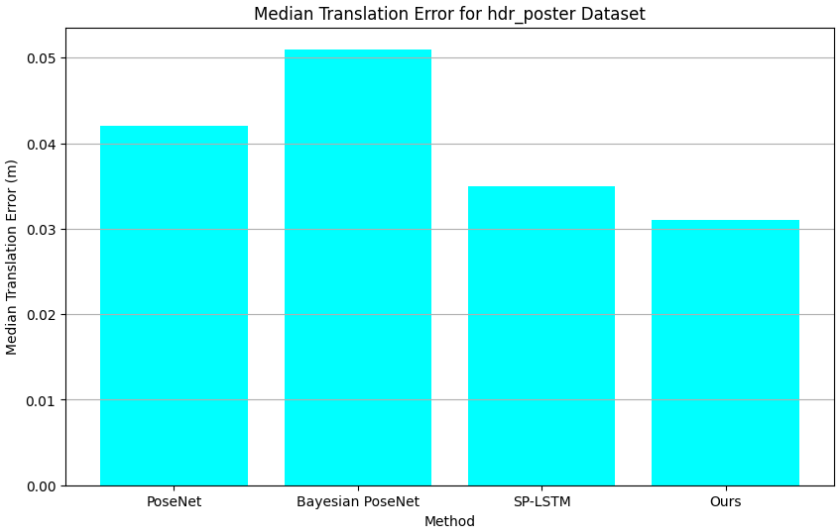

Figure 3: Median error of translation for the hdr_poster dataset.

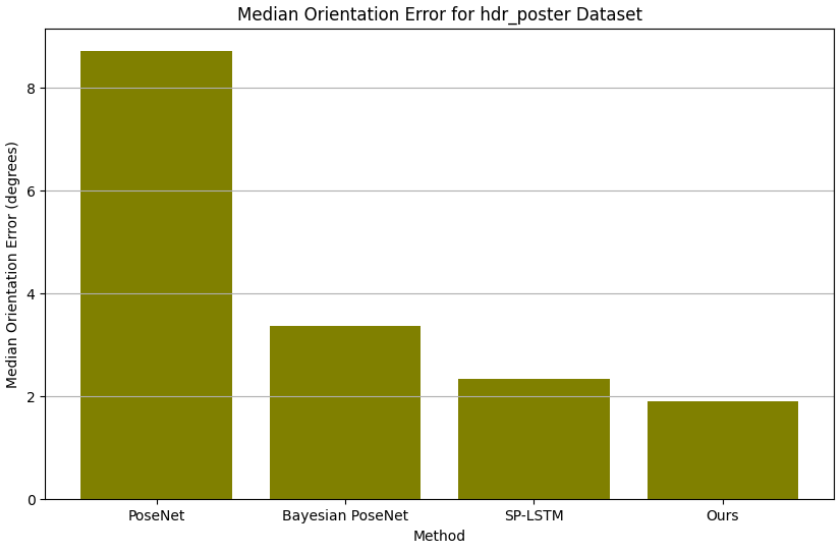

Figure 4: Median error of rotation for the hdr_poster dataset.

els namely PoseNet[12], Bayesian PoseNet [11] and SP-LSTM[25] using CNN and LSTM.

5.0.2 *Results with Random Split*

The outcomes highlighted in Table 1 were garnered utilizing a random split methodology. We deployed 6 sequences (**shapes rotation, box translation, shapes translation, dynamic 6dof, hdr poster, poster translation**) for this set of experiments. Across all these sequences, our model consistently exhibits the least average and mean errors. It reaches an average median error of 0.031m and 2.198°in all sequences of the real dataset, which surpasses the most recent method, SP-LSTM, that records results of 0.036m and 2.341°, PoseNet and Bayesian PoseNet outcomes were 0.231m, 7.455°and 0.241m, 7.593°, respectively.

5.0.3 *Results with Novel Split*

The comparison outcomes from the novel split, as delineated in Section 4.1, are presented in Table 2. The table indicates that the novel split strategy is more challenging than the random split approach as all methods reported higher error rates. We employed three sequences from the shapes scene (**shapes rotation, shapes translation, shapes 6dof**) for this particular split. The outcomes obtained through our method outperform those garnered with state-of-the-art techniques. We attained an average median error of 0.061m and 4.713°across all sequences of the real dataset, in comparison to the SP-LSTM method which resulted in 0.065m, 5.012°. Furthermore, PoseNet and Bayesian PoseNet reported errors of 0.240m, 11.067°and 0.234m, 10.975°respectively.

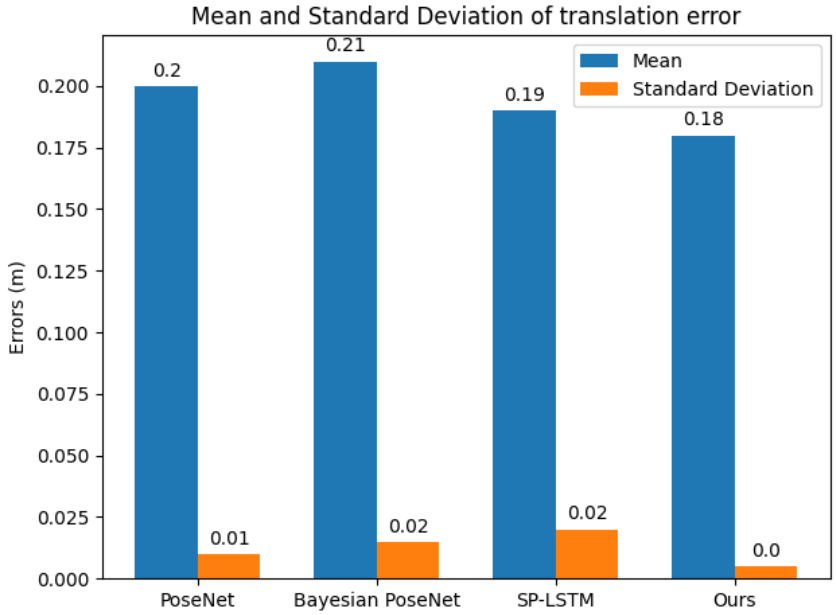

Figure 5: Distribution of translation error.

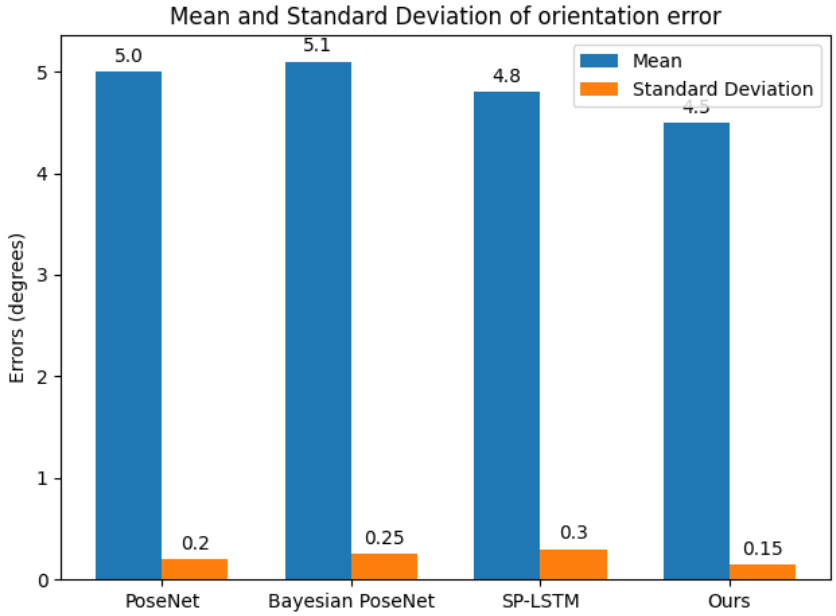

Figure 6: Distribution of rotation error.

We should note that in the novel split strategy, the testing set comprises the final 30% of the event images. This means that we lack the "neighborhood" relationship between the training and testing images. Conversely, in the random split, the testing images could be in close proximity to the training images since the images are randomly chosen from the entire sequence for training/testing.

One of the key advantages of our method is its ability to capture long-range dependencies in the event data using the Transformer encoder, which contributes to the improved performance. We also observe that the ResNet50 backbone plays a crucial role in extracting relevant spatial features from the event data. This is evidenced by the model's robustness against rapid motion and high-dynamic-range scenes, which are typically challenging for traditional frame-based vision algorithms. In future work, we aim to further improve the model's performance by investigating other feature extraction backbones and transformer architectures, and by exploring data augmentation techniques tailored to event data.

6 CONCLUSION

In this work, we presented a novel method for 6-DOF pose estimation using event camera data, combining a pretrained ResNet50 for feature extraction with a custom transformer encoder for capturing the inherent spatial and temporal dependencies. Our method provides an efficient way to handle the non-sequential changes in the scene that are common in event data. Through comprehensive experiments, we demonstrated that our method outperforms the current state-of-the-art techniques in both random and novel split strategies. Despite the success of our method, there are still opportunities for further improvement. For future work, we aim to explore additional ways of enhancing the transformer's capability to better capture the complexity of the spatial-temporal event data.

7 REFERENCES

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[2] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[3] Jörg Conradt, Matthew Cook, Raphael Berner, Patrick Lichtsteiner, Rodney J Douglas, and Tobi Delbruck. A pencil balancing robot using a pair of aer dynamic vision sensors. In *2009 IEEE International Symposium on Circuits and Systems*, pages 781–784. IEEE, 2009.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[5] Timothy Dozat. Incorporating nesterov momentum into adam. 2016.

[6] Andreas Eitel, Jost Tobias Springenberg, Luciano Spinello, Martin Riedmiller, and Wolfram Burgard. Multimodal deep learning for robust rgb-d object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 681–687. IEEE, 2015.

[7] Guillermo Gallego and Davide Scaramuzza. Accurate angular velocity estimation with an event camera. *IEEE Robotics and Automation Letters*, 2(2):632–639, 2017.

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[9] Xian-Feng Han, Hamid Laga, and Mohammed Bennamoun. Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1578–1604, 2019.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *2016 IEEE international conference on Robotics and Automation (ICRA)*, pages 4762–4769. IEEE, 2016.

[12] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015.

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155–166, 2009.

[15] Ming Li, Ruizhi Chen, Xuan Liao, Bingxuan Guo, Weilong Zhang, and Ge Guo. A precise indoor visual positioning approach using a built image feature database and single user image from smartphone cameras. *Remote Sensing*, 12(5):869, 2020.

[16] Ming Li, Jiangying Qin, Deren Li, Ruizhi Chen, Xuan Liao, and Bingxuan Guo. Vnlstm-posenet: A novel deep convnet for real-time 6-dof camera relocalization in urban streets. *Geo-spatial Information Science*, 24(3):422–437, 2021.

[17] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 1449–1457, 2015.

[18] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[19] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van

Der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European conference on computer vision (ECCV)*, pages 181–196, 2018.

[20] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5419–5427, 2018.

[21] Elias Mueggler, Basil Huber, and Davide Scaramuzza. Event-based, 6-dof pose tracking for high-speed maneuvers. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2761–2768. IEEE, 2014.

[22] 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. *The International Journal of Robotics Research*, 36(2):142–149, 2017.

[23] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015.

[24] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017.

[25] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[26] Etienne Perot, Pierre de Tournemire, Davide Nitti, Jonathan Masci, and Amos Sironi. Learning to detect objects with a 1 megapixel event camera. *Advances in Neural Information Processing Systems*, 33:16639–16652, 2020.

[27] Chao Qu, Shreyas S Shivakumar, Ian D Miller, and Camillo J Taylor. Dsol: A fast direct sparse odometry scheme. *arXiv preprint arXiv:2203.08182*, 2022.

[28] Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. 2017.

[29] 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.

[30] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1744–1756, 2016.

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

[32] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[33] Ahmed Tabia, Fabien Bonardi, and Samia Bouchafa. Deep learning for pose estimation from event camera. In *2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–7. IEEE, 2022.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.