All-in-Focus Imaging of Dynamic Scenes with an Event-Based Focal Stack

Akshaj Gaur*
Kalonji Harrington*
Ji-Ze Jang*
agaur@umd.edu
kharring@umd.edu
gjang@umd.edu
University of Maryland, College Park
College Park, Maryland, USA

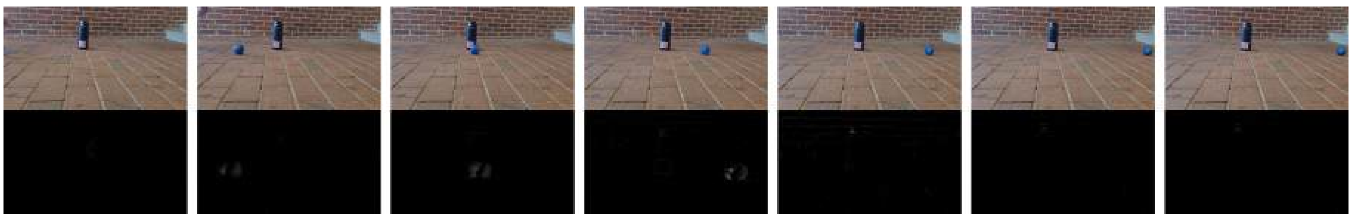


Figure 1: A seven-frame comparison of the scene between the RGB frames and the event-based focal stack. The top layer depicts the scene itself, while the bottom layer represents the event focal stack from the event-based camera during a focal sweep. Each frame is a slice of summed events during a 10 millisecond time window.

Abstract

Lens-based imaging presents several challenges for all-in-focus imaging, such as ensuring that all parts of the scene are in focus. While frame-based methods have been proposed for all-in-focus imaging, they often result in low temporal resolution due to frame rate limitations. Event focal stacks have offered promise in the field, but current works are limited to static scene reconstruction. We propose a novel all-in-focus imaging method for dynamic scenes using an event-based focal stack. In this paper, we discuss our preliminary experiments for dynamic scene reconstruction. We found promising initial results for grayscale reconstruction of dynamic scenes with our proposed divide-and-conquer method.

Keywords

all-in-focus, event-based vision, computational imaging

1 Introduction

Lens-based imaging relies on curved glass to bend light to create sharp images. However, the main limitation of this method is that only parts of the scene at a particular distance from the sensor are fully in focus. The remainder of the scene suffers from focus deblur, making sharp images of scenes with large depth disparities impossible with most conventional sensors. This is desirable in some cases for artistic reasons, but is often undesirable when full information of the scene is important. All-in-focus imaging removes this depth-dependent blurring to retain full scene information at every distance.

Previous all-in-focus (AIF) imaging methods have relied on the target scene remaining static throughout the exposure, limiting their applications. Dynamic all-in-focus imaging presents the unique challenge of having different scenes between images in the focal stack. Because of this, it cannot be assumed that the same pixels in different stack images correspond to the same scene point, so the weighted sum approach is no longer feasible. This also introduces the concern that some scene points may never be in focus, as they become occluded by a dynamic object.

Other approaches have attempted to tackle the dynamic all-in-focus problem through a frame-based camera, but these methods also have significant drawbacks. First, frame-based approaches often lead to a lower temporal resolution due to the frame rate limits of the camera. In dynamic scenes, this can lead to motion blur and misalignment between frames. Second, capturing frames in succession results in a lower exposure time, which can hurt performance in darker conditions and signal-to-noise ratio. In addition, frame-based images are often prone to camera shake, which exacerbates the misalignment in the focal stack and blurs the final image.

In recent years, event-based cameras have presented several benefits to address these shortcomings. Event-based cameras not only capture high-speed motion and achieve real-time data collection, but they also overcome depth-of-field limitations to construct all-in-focus images.

The goal of this project is to use an event-based focal stack to perform dynamic all-in-focus imaging. The asynchronous detection and high temporal precision of an event-based vision sensor, also known as a dynamic vision sensor (DVS), allow it to quickly capture information from focal sweep while also providing rich motion information for tracking. We propose a divide-and-conquer approach to dynamic scene reconstruction. In our method, we reconstruct static intensity images from event camera data based on existing static scene all-in-focus methods. But we extend these

*Names are ordered alphabetically

previous works to dynamic scenes by manually tracking objects across frames.

To summarize, our contributions are as follows:

- We propose a novel method to extend existing static scene reconstruction approaches to *dynamic* scenes (i.e., a scene with a static background and a moving foreground object)
- We collected our own event focal stack (EFS) for the dynamic scene reconstruction from manual and continuous focus adjustment
- We assess different methods of measuring focus from an event focal stack
- We propose and attempt to implement a method for object detection during a focal sweep

2 Related Work

We provide an overview of existing approaches for reconstructing an all-in-focus (AIF) image from an image focal stack using frame-based camera. Then, we discuss prior works and limitations of depth-from-defocus using coded apertures, which has been extensively studied over the past several decades. Finally, we describe a more recent approach of reconstructing all-in-focus image from an event focal stack, and how our work builds on this line of research to reconstruct an all-in-focus *dynamic* scene.

2.1 All-in-Focus from Image Focal Stack

Image focal stack methods capture multiple images at different focus distances, either by capturing multiple images sequentially [14] [3] or using light fields [9]. The former requires the scene to remain static, while the latter sacrifices spatial resolution.

Huang et al. (2022) compared the performance between focal stack and lightfield cameras. The study focused on depth estimating using two methods: creating the focal stack from publicly available lightfield datasets and experimental capture. But neither of the methods used are applicable to dynamic scenes. Similarly, Suwajanakorn et al. (2015) explored the possibility of depth from defocus using mobile phones and handheld cameras. The study had limited scope into dynamic scenes as it focused on three scenarios: static scenes, scenes with 0.25-inch camera motion and scenes with one-inch camera motion. These findings offer limited applications to dynamic all-in-focus imaging. Another limitation of image-based focal stack methods is that they are often prone to camera shake. This can lead to misalignment within the focal stack and blurring in the final all-in-focus image.

Wang et al. (2012) [16] proposed a deep-learning pipeline for autofocus and all-in-focus imaging with applications for dynamic scenes. While the approach outperformed traditional static scene reconstruction methods, it had limitations for dynamic scene reconstruction due to low temporal resolution stemming from a frame-based camera.

2.2 Depth from Defocus and Coded Apertures

Coded apertures can create an image focal stack from a single image with full spatial resolution [5]. Using the defocus cues from the coded aperture, refocusing an image is just a spatially varying deconvolution problem. However, the defocus kernel can be difficult to estimate, making the refocused images and resulting all-in-focus images inaccurate. Previous approaches, including that of Levin et al. (2007) [6], required an exact calibration for the blur filter across the depth values. This poses challenges for extreme defocus values as blur can't be accurately inverted. Other coded aperture approaches, such as Zhou et al. (2009) [18], used a pair of apertures to construct the all-in-focus image. While the approach led to stronger results in the presence of weak textures, it had significant blurring along occlusion boundaries.

End-to-end learning methods have shown great success [4] [11], but create artifacts when deblurring high-frequency components. Many of these methods also require the fabrication of coded apertures or wavefront coding optimized for the task.

For example, Lee et al. (2021) proposed a defocus deblurring framework that predicts per-pixel deblurring filters. But the approach struggles with large blurs and blurs in irregular shapes due to a lack of training data. More recent work, including Ruan et al. (2022), had similar shortcomings in deblurring irregular shapes.

2.3 All-in-Focus from Event Focal Stack

Recently, neuromorphic vision systems have been used to create focal stacks for all-in-focus imaging [8] [15].

Lou et al. (2023) used an event focal stack (EFS) for all-in-focus imaging. The paper employs the golden-section search to select the refocusing timestamps and reconstruct the corresponding refocused images to create the focal stack. Using the timestamps's neighboring events, they merge the focal stack with proper weights and construct the final all-in-focus image. After finding the timestamps, the approach uses EvRefocusNet and EvMergeNet to retrieve the image. However, the event focal stack was captured manually, which was not fast enough to record scenes with object motion, so the method is not viable for dynamic scenes. Bao et al. (2023) [1] presented an alternative timestamp finding algorithm that is faster and more accurate than the golden search used by Lou et a

Teng et al. (2024) attempted a similar approach by using a neuromorphic camera for hybrid all-in-focus imaging. But the approach can only capture gray-scale changes in intensity and is still not viable for dynamic scenes.

Multiple studies [10] [2] also allude to the use of a liquid lens to speed up the manual focal sweep and allow for potential applications to dynamic scenes. But currently, liquid lenses are limited in their applications and are not viable for day-to-day imaging tasks. For example, Ralph et al. (2024) used a liquid lens for space image, which resulted in a higher focus.

2.4 Event-Based Image Reconstruction

Event-based cameras have also been used for more general reconstruction techniques in both static and dynamic scenes [12] [17]. Scheerlinck et al. (2020) proposed an efficient neural network architecture for video reconstruction, which had similar performance to larger models. Other methods, including Zhang et al. (2020), achieved similar results with a convolution neural network method. But neither of the works focused on the all-in-focus imaging problem.

Figure 2: All-in-focus (AIF) static reconstruction using event focal stack and guided by an RGB image [8]. Left: all-in-focus image. Right: weights used to merge RGB focal stack to create the all-in-focus image.

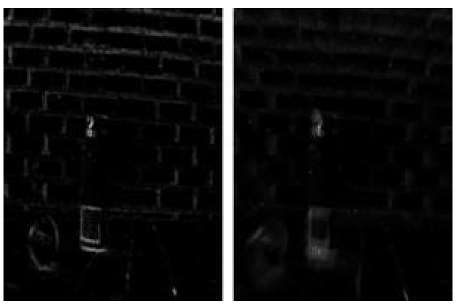


Figure 3: Our static reconstructions from event focal stack, accumulating events using different sized time windows. Left: narrow window. Right: wide window. Current results are still in grayscale.

2.5 Liquid Lenses

Liquid lenses offer several benefits for event-based all-in-focus methods due to rapid focal adjustment, which can help dynamic focus applications. But recent liquid lens research has focused on more ambitious applications, such as space imaging [10], while other works have focused on static, 3D scenes with biomedical imaging applications [2].

3 Methods

We propose a divide-and-conquer method for reconstructing an all-in-focus dynamic scene. First, we perform static reconstruction of each frame of dynamic scene (i.e., ball rolling across the floor in front of a static brick wall and water bottle). Then, we manually track the moving object (i.e., ball) across the frames to recover its motion. Our goal is to combine the high-quality per-frame static reconstructions with tracking to reconstruct an all-in-focus dynamic scene. Note that our current results are grayscale and require manual tracking of objects. We discuss our next steps and plan to extend our current work in Section 5.

3.1 Static Image Reconstruction

Reconstructing motion has been a longstanding challenge in computer vision and computational imaging. Therefore, we process the dynamic scene by separately optimizing for per-frame (spatial) and between-frame (temporal) reconstruction qualities, then combining them to recover the dynamic scene. For static reconstruction, we are inspired by the results in [8], which can be found in Figure 2. First, we plotted binary polarity values across all timestamps of our event focal stack to determine the time frame during which most events occurred. Similar to [8], we use a patch-based iterative golden-section search to determine the optimal timestamps for which the scene is most in focus. Then, we accumulate the events in the selected temporal windows to obtain the reconstructed image. Our reconstruction of a static scene sweep is shown in Figure 3.

3.2 Measuring Focus

As in [8], the event-based focal sweep encodes texture information for the continuous focus depths covered by the sweep. In order to obtain an appropriate focal stack from this data, a series of timestamps must be selected in such a way that each region of the scene is in focus at at least one timestamp. We consider two possible methods of measuring focus from events: event rate and reconstructed sharpness.

Using the polarity-based event-rate evaluation method designed for autofocus in [1], focus can be evaluated for the entire scene. However, for all in focus there will be different optimal focus timestamps for different parts of the scene. This could result in the method prioritizing a particular part of the scene, or averaging the best focus measures for different regions.

To adapt the method for this use case, we propose calculating the event-rate for each region of the scene individually to estimate its focus value. Additionally, adding a dynamic object into the process has the potential to reduce this method's viability due to the additional events from the motion. We will evaluate the focus measure on both the static objects and dynamic objects in the scene.

A more common method of evaluating focus is to examine the sharpness of a reconstructed image, as in the previous all-in-focus paper [8]. This method is more robust and allows the use of various well-established focus measures for images. However, it does require that the DVS and RGB camera have the same field of view, usually by using a beam splitter. In addition, the slower speed of the camera's frame rate can make it more difficult to align the dynamic object.

Rather than having multiple RGB images from a simultaneous video of the dynamic scene that may be difficult to align spatially and temporally, we instead use a single static image with all objects (which may be a single frame from a video). The dynamic object is identified in the RGB image and tracked in the event focal stack so that during image reconstruction, only the events corresponding to the object are used.

3.3 Object Tracking

Due to the high volume of events that occur during a focal sweep, performing feature detection and tracking from events alone is infeasible using traditional methods. Given this, image-based detection on accumulated frames is necessary for this task. This method also comes with the added benefit of making it easier to match the objects found in events to objects identified in the RGB image.

For our testing, we assume the shape of the object is known, potentially by identifying objects in the RGB scene. Given this shape, the Hough Transform can be used to identify the object in an accumulated frame. If tracking through accumulated frames is unsuccessful, we also implement a hand-tuned tracking box that can isolate the events for a specific recording. It is a single rectangle that moves at a particular speed across the frame, assuming the object moves at a constant velocity.

3.4 AIF Image Construction

For our focal sweep, we fix the lens to be as telephoto as possible with the aperture fully open, adjusting the focal length to obtain a continuous sweep of focus distances. A full sweep is not conducted, as the object distances are close enough that a partial sweep is enough to ensure that every object is in focus at some point. The RGB image is captured separately and focused at an arbitrary distance.

Once identified, the events corresponding to the dynamic object are stored separately. This allows the reconstruction of the static scene to only use events that are caused by the focal sweep. Reconstruction for this part of the scene is similar to previous methods, except that events from parts of the scene where the object was found will be missing. This risks parts of the scene never being in focus during the sweep, but that risk can be eliminated if multiple sweeps are done during the object's motion and the object.

To construct an all-in-focus image from this separated event focal stack, we divide the problem into constructing an all-in-focus image of the static background and constructing an all-in-focus image of the moving object. For the static background, we can utilize the same pretrained neural networks from Lou et al. [8] to reconstruct the image. For the dynamic background, the same can be done for only the part of the RGB image that contains the object, resulting in an in-focus image of the object. Then, these two images are merged using a mask of where the object is present.

4 Experiments

4.1 Recording Setup

For the initial trials of the all-in-focus system, the event focal stack was captured using a manual continuous focus adjustment. Although the adjustments are performed by hand, making the sweep relatively slow, they are still fast enough to complete a sweep while the object is still in frame.

We recorded an EFS and RGB image with a textured background as shown in Figure 4. There is one dynamic object and one static object in the scene, with the dynamic object being the closest so it is never occluded. The RGB image was focused such that only the floor in the foreground was in focus, with both objects remaining blurry.

The current setup and lens used for the EFS does have some flaws, particularly with the field of view. The lack of a beam splitter means that the field of view for the sensors cannot be kept the same. In addition, adjusting the focal length causes a change in the field of view during the sweep.

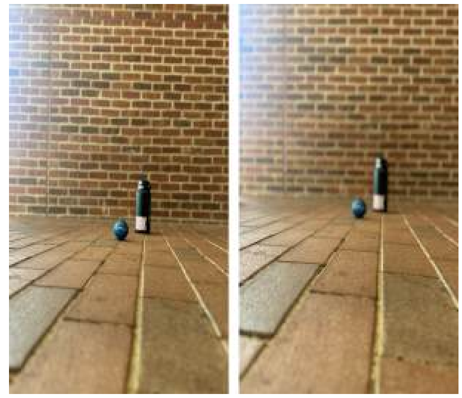


Figure 4: The static scene experimental setup with defocus blur at different depths in the scene. Left: the ball, water bottle, and brick wall are in focus while the floor is defocused. Right: parts of the floor is in focus while the ball, water bottle and brick wall are defocused.

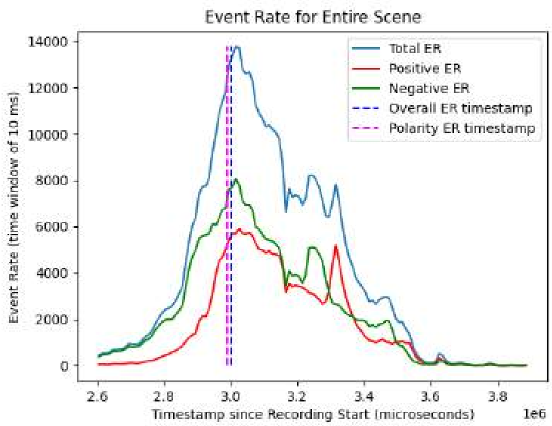
4.2 Measuring Focus

Full Scene. The event-rate measurement was designed to evaluate the sharpness of a full scene. We first examined the method's results when applied over all events in the sweep. The results in Figure 5 show that there was very little different between the timestamps from the overall event-rate method and the polarity-based method. The slice of events shows that the background wall is most in focus using this evaluation metric.

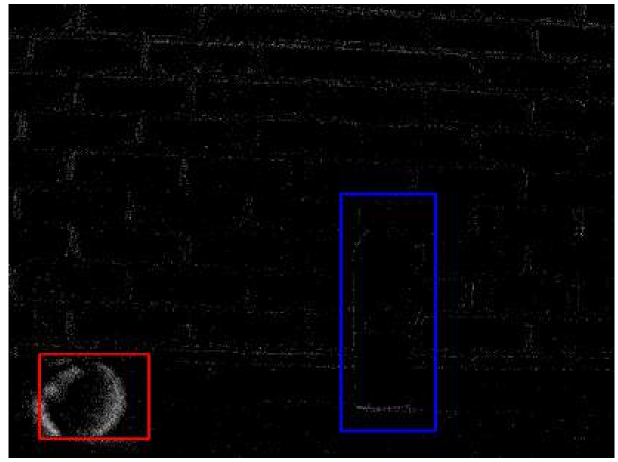
Static Object. As a preliminary test of viability of focusing different regions, the static water bottle is manually isolated in the event stream. The same event-rate test is run assuming these events represent the entire scene. As seen in Figure 5c, the event-rate shows a very different structure due to the ball not being present during most of the sweep, but eventually entering. Despite the irregular structure due to the added events, the selected timestamp is later, when the focus distance is nearer, as expected. The edges of the bottle are much sharper, with the text of the bottle being more visible.

Image Sharpness Methods. The image sharpness method requires the reconstruction of individual patches within an image based on the EFS. We attempted to use the pre-trained refocusing network for this task, expecting that for the dynamic parts of the scene it would perform poorly, but for the static parts it may perform adequately.

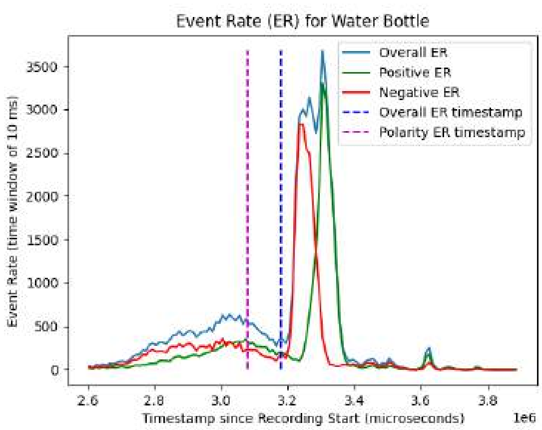
However, with further testing, we found that while the dynamics in the scene certainly degraded performance, there were other major factors that limited the methods applicability. To test this, we ran an experiment with a static focal sweep. The results in Figure 6 show that there are artifacts around the top-left of the image, which is also where all of the patches were identified. This is because the resolution of the sensor is much higher than the DVS, and the fields of view are not aligned. As a result, the events are not properly mapped to the RGB pixels, and the reconstruction fails.



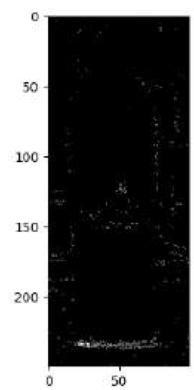
(a) Plot of event rate for the entire scene. Comparison between polarity-based event-rate from [1] and overall event rate from [13].



(b) Slice of events around the optimal focus timestamp from the Polarity ER method. The wall is most in focus for the chosen timestamp.



(c) Plot of event rate for the entire scene. Comparison between polarity-based event-rate from [1] and overall event rate from [13].



(d) Slice of events around the optimal focus timestamp from the Polarity ER method. The bottle is more in focus, with text more visible, but not perfectly in focus.

Figure 5: Results of using events-rate method for focus measure. Using it on the overall scene as in 5a and 5b results in the largest feature (the wall) being most in focus. Only using events from a specific region as in 5c and 5d selects the timestamp that is most in focus for the object in that region. The different in structure between the graphs is likely due to the object's motion, with the spike in 5a the result of the object rolling into the region.

4.3 Object Tracking

The original goal of the tracking was to use an accumulated frame to detect objects in the event stream. OpenCV's Hough circle transform was our candidate feature detector since we knew the shape of the object to be a cirlce a priori. However, likely due to the large volume of events or a package issue, the accumulator interface provided by the sensor's manufacturer is not usable. The pre-implemented event-based feature detectors also provided by the package were not successful.

In lieu of implementing the accumulation, we opted to begin using manual tracking of the dynamic object. This simply consisted of initializing a tracking box to the location of the ball and moving it at a constant velocity that followed the ball. This tracking is recording-specific, as the ball does not always enter the frame at the same time or follow the exact same path, but does allow for some preliminary separated data.

4.4 Static Image Reconstruction

Our static reconstruction results can be found in Figure 3. We find that using polarity data to determine the optimal time window for accumulating events is a crucial step for good reconstructions: the more precise and narrow our time window is, the better the reconstruction. Integrating events over a large time window not only captures the motion of the focal sweep, leading to a blurry

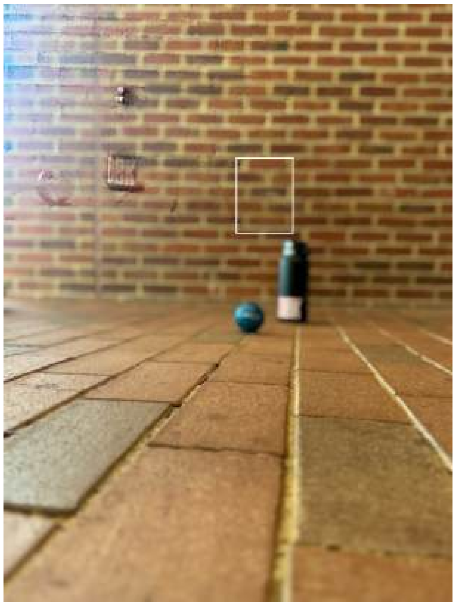


Figure 6: The image reconstruction obtained from Lou et al.'s image sharpness method meant to determine when the given path is most in focus [8]. Due to the field of view and resolutions being mismatched, there is artifacting from the objects concentrated in the top-left corner of the reconstructed image.

image, but also the inherent noise that is detected by an event camera sensor.

Note that, since an event-based system only detects intensity changes, our reconstruction is grayscale and an RGB image would be needed to recover the full color space of the scene.

4.5 AIF Image Reconstruction

Due to the errors with field of view when evaluating image sharpness, the original proposed pipeline is not feasible. That method requires the use of a neural network to generate an image focal stack, but the resulting focal stack would inevitably have artifacts similar to the patches in Figure 6.

5 Discussion

In this paper, we proposed a novel method for all-in-focus imaging for dynamic scenes using an event-based focal stack. We employed a divide-and-conquer method for the reconstruction as we used existing static reconstruction methods and combined them with manual tracking for dynamic scenes Our preliminary results and experiments are promising for furthering the use of event focal stacks for dynamic scene reconstruction.

5.1 Limitations

The current work has multiple limitations that can be addressed. The first is the slow speed of the focal sweep, which both increases the distance traveled by the object during the sweep and puts a limit on how fast the dynamic object can be moving. We also observed that the mismatched field of view between the camera and the DVS made measuring focus through reconstruction infeasible. Ensuring they have the same field of view with a beam splitter would be ideal, but simply cropping the RGB image to have approximately the same field of view as the DVS would potentially help achieve better reconstructions.

A more fundamental limitation is that the object must remain at virtually the same depth throughout the focal sweep to ensure it is in focus at at least one timestamp. The method also does not accommodate camera motion, for a similar reason, as the new scene the camera will encounter during the motion may never be in focus during the sweep.

5.2 Future Work

Our full proposed method has yet to be fully implemented and tested. Modifications to the AIF pipeline to separately reconstruct the dynamic and static elements of the scene still need to be completed, and issues caused by the field of view need to be resolved.

Currently, our divide-and-conquer approach reconstructs each frame in the event focal stack as a static scene, and manually tracks the moving object across all frames. Adding automatic object tracking would require a fast implementation of event accumulation for traditional object-detection methods to be used. Implementing and evaluating this tracking method would be crucial to making the proposed method viable.

However, our goal is to reconstruct an all-in-focus dynamic scene, where we can clearly recover the continuous motion of the moving object as if it were recorded from a regular image-based camera. To improve the quality of motion reconstruction, we hope to experiment with more complex hardware setup, such as using a beam splitter [8] to simultaneously capture event-based and frame-based images, as well as using a liquid lens [7] to enable fast focusing capability. By leveraging more advanced hardware, we hope to enhance the current method to increase its flexibility (e.g., handling high dynamic range motion) and enable it to operate effectively under extreme conditions (e.g., low-light settings).

While deep learning approaches impose additional requirements such as training data collection, we believe that the results in [8] demonstrate neural networks as a promising solution for encoding continuous and richer information that can make our framework more flexible. Collecting data with many different objects with different backgrounds to act as training data for a similar network may be the best path forward for the refocusing aspect. Adding a mechanized or liquid lens could greatly improve the efficiency of collecting this data. This would also remove the need for explicit tracking, which could remove a point of failure of the method. Comparing a full deep learning approach to our current proposed pipeline (once completed) would be also be a great avenue for future exploration.

Acknowledgments

A huge thank you to Dr. Christopher Metzler for providing supplies and support to make this project successful.

Team Contributions:

- Akshaj: data collection, static reconstruction, writing
- Kalonji: project ideation, data collection, event rate and image sharpness tests, object tracking, writing

- Ji-Ze: project ideation, data collection, static reconstruction, baseline [8], writing

References

[1] Yuhan Bao, Lei Sun, Yuqin Ma, Diyang Gu, and Kaiwei Wang. 2023. Improving fast auto-focus with event polarity. Optics Express 31, 15 (July 2023), 24025. https://doi.org/10.1364/oe.489717

[2] Huayu Cheng, Lihui Wang, Satoshi Tabata, Yuan He, Yan Hu, Jiang Liu, and Zhiwei Mou. 2024. High-speed all-in-focus 3D imaging method based on liquid lens focus scanning. Appl. Opt. 63, 21 (Jul 2024), 5602–5610. https://doi.org/10.1364/AO.523864

[3] Zhengyu Huang, Jeff Fessler, and Theodore Norris. 2022. Focal stack camera: depth estimation performance comparison and design exploration. Optics Continuum 1 (09 2022). https://doi.org/10.1364/OPTCON.472819

[4] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. 2021. Iterative Filter Adaptive Network for Single Image Defocus Deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[5] Anat Levin, Rob Fergus, Frédo Durand, and William T. Freeman. 2007. Image and depth from a conventional camera with a coded aperture. 26, 3 (July 2007), 70–es. https://doi.org/10.1145/1276377.1276464

[6] Anat Levin, Rob Fergus, Frédo Durand, and William T. Freeman. 2007. Image and depth from a conventional camera with a coded aperture. In ACM SIGGRAPH 2007 Papers (San Diego, California) (SIGGRAPH '07). Association for Computing Machinery, New York, NY, USA, 70–es. https://doi.org/10.1145/1275808.1276464

[7] Carlos Lopez and Amir Hirsa. 2008. Fast focusing using a pinned-contact oscillating liquid lens. Nature Photonics 2 (2008).

[8] Hanyue Lou, Minggui Teng, Yixin Yang, and Boxin Shi. 2023. All-in-Focus Imaging From Event Focal Stack. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 17366–17375.

[9] Liu Peng and Liu Dijun. 2013. All-in-Focus Image Reconstruction Based on Plenoptic Cameras. In 2013 Seventh International Conference on Image and Graphics. 612–617. https://doi.org/10.1109/ICIG.2013.127

[10] Nicholas Ralph, Darren Maybour, Alexandre Marcireau, Nik Dennler, Sami Arja, Nimrod Kruger, and Gregory Cohen. 2024. Active Neuromorphic Space Imaging and Focusing using Liquid Lenses. https://doi.org/10.36227/techrxiv.173152542.28624013/v1

[11] Lingyan Ruan, Bin Chen, Jizhou Li, and Miuling Lam. 2022. Learning to Deblur using Light Field Generated and Real Defocus Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 16304–16313.

[12] Cedric Scheerlinck, Henri Rebecq, Daniel Gehrig, Nick Barnes, Robert Mahony, and Davide Scaramuzza. 2020. Fast Image Reconstruction with an Event Camera. 156–163. https://doi.org/10.1109/WACV45572.2020.9093366

[13] Lin Shijie, Zhang Yinqiang, Yu Lei, Zhou Bin, Luo Xiaowei, and Pan Jia. 2022. Autofocus for Event Cameras. In The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

[14] Supasorn Suwajanakorn, Carlos Hernandez, and Steven M. Seitz. 2015. Depth from focus with your mobile phone. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 3497–3506. https://doi.org/10.1109/CVPR.2015.7298972

[15] Minggui Teng, Hanyue Lou, Yixin Yang, Tiejun Huang, and Boxin Shi. 2024. Hybrid All-in-Focus Imaging From Neuromorphic Focal Stack. IEEE Transactions on Pattern Analysis and Machine Intelligence 46, 12 (2024), 10124–10137. https://doi.org/10.1109/TPAMI.2024.3433607

[16] Chengyu Wang, Qian Huang, Ming Cheng, Zhan Ma, and David J. Brady. 2021. Deep Learning for Camera Autofocus. IEEE Transactions on Computational Imaging 7 (2021), 258–271. https://doi.org/10.1109/TCI.2021.3059497

[17] Zelin Zhang, Anthony Yezzi, and Guillermo Gallego. 2022. Formulating Event-based Image Reconstruction as a Linear Inverse Problem with Deep Regularization using Optical Flow. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022), 1–18. https://doi.org/10.1109/tpami.2022.3230727

[18] Changyin Zhou, Stephen Lin, and Shree Nayar. 2009. Coded Aperture Pairs for Depth from Defocus. Proceedings of the IEEE International Conference on Computer Vision, 325–332. https://doi.org/10.1109/ICCV.2009.5459268