Journal of Robotics and Control (JRC)
Volume 5, Issue 4, 2024
ISSN: 2715-5072, DOI: 10.18196/jrc.v5i4.21930

A Recurrent Deep Architecture for Enhancing Indoor Camera Localization Using Motion Blur Elimination

Muhammad S. Alam [1*], Farhan B. Mohamed [2], Ali Selamat [3], AKM B. Hossain [4]
[1,2,4] Department of Emergent Computing, School of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia
[2] Media and Game Innovation Centre of Excellence (MaGICX), Universiti Teknologi Malaysia, Johor Bahru, Malaysia
[3] Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia
[1,4] Department of Computer Science and Artificial Intelligence, College of Computing and Information Technology, University of Bisha, Bisha, Saudi Arabia
Email: [1] shamsul20@graduate.utm.my, [2] farhan@utm.my,
[3] aselamat@utm.my, [4] k.m.a@graduate.utm.my
*Corresponding Author

***Abstract*—Rapid growth and technological improvements in computer vision have enabled indoor camera localization. The accurate camera localization of an indoor environment is challenging because it has many complex problems, and motion blur is one of them. Motion blur introduces significant errors, degrades the image quality, and affects feature matching, making it challenging to determine camera pose accurately. Improving the camera localization accuracy for some robotic applications is still necessary. In this study, we propose a recurrent neural network (RNN) approach to solve the indoor camera localization problem using motion blur reduction. Motion blur in an image is detected by analyzing its frequency spectrum. A low-frequency component indicates motion blur, and by investigating the direction of these low-frequency components, the location and amount of blur are estimated. Then, Wiener filtering deconvolution removes the blur and obtains a clear copy of the original image. The performance of the proposed approach is evaluated by comparing the original and blurred images using the peak signal-to-noise ratio (PSNR) and structural similarity index(SSIM). After that, the camera pose is estimated using recurrent neural architecture from deblurred images or videos. The average camera pose error obtained through our approach is (0.16m, 5.61°). In two recent research, Deep Attention and CGAPoseNet, the average pose error is (19m, 6.25°) and (0.27m, 9.39°), respectively. The results obtained through the proposed approach improve the current research results. As a result, some applications of indoor camera localization, such as mobile robots and guide robots, will work more accurately.**

***Keywords*—Camera Pose Estimation; Indoor Camera Localization; Indoor Robot Navigation; Motion Blur; RRN; SLAM.**

I. INTRODUCTION

Indoor camera localization is essential to identify the position and orientation of a camera within an environment relative to a specific object. This domain is fundamental to computer vision research, emphasizing indoor robot navigation. The primary objective is to accurately determine the camera's position, or its 'pose', within a physical space. Such localization techniques are crucial for various robotic tasks, including navigation, scene reconstruction, and object recognition. Recent advances have significantly improved the accuracy and robustness of indoor camera localization. Despite these advances, indoor robot navigation and surveillance applications demand more precise camera pose prediction. One of the major issues in improving camera localization in indoor environments is motion blur. Motion blur is one of the major obstacles to more accurate camera localization. It degrades the quality of captured images and reduces the accuracy of camera localization. Motion blur elimination is essential to improve localization performance.

Motion blur is a typical problem in indoor camera localization when capturing images of moving objects or in poor light. The accuracy of localization algorithms can be significantly decreased by producing distortions and ambiguities in the collected images. Motion blur must be addressed for camera localization systems to operate more effectively. The appropriate deep-learning architecture is essential for precise and trustworthy indoor camera localization. Traditional methods frequently use single-image algorithms, which may not fully exploit the temporal information present in the image sequences. It is possible to improve localization outcomes using temporal data collected over several subsequent frames as helpful cues for motion estimates and blur removal. It is required to sophisticated image deblurring techniques that successfully restore the sharpness and clarity of blurred images to remove motion blur.

In robotics and computer vision, indoor camera localization is a significant problem [1], [2]. Various methods have been developed for indoor camera localization [3]–[5], which is

crucial for applications such as indoor robot navigation, SfM, and SLAM. A method involves point-based techniques that use image descriptors and 3D scene point clouds obtained from SfM to establish a camera pose based on 2D-3D matches. However, this method may not be accurate in some situations, such as when motion blur [6]–[9] is present. To address the motion blur problem, researchers have proposed a machine learning technique [10]–[14], which estimates camera pose based on the predicted 3D locations of an input image. However, depth maps must be matched with input images during training, which must be performed within a limited time. As a result, the training architecture uses a mapping from pixel to pose [15] dependent on the coordinates system. Developing a comprehensive indoor positioning system has been a challenging research area for many years, with precise pose data essential for various applications, such as autonomous robot navigation [16]–[18].

We aim to develop a deep neural network architecture that improves indoor camera localization performance by combining motion blur removal processes and recurrent neural networks. This method demonstrates how the accuracy and robustness of localization can be increased, especially in difficult motion blur situations. In this research study, we propose an innovative recurrent deep architecture for indoor camera localization that removes motion blur from an image or sequence of images and improves localization performance. The recurrent deep architecture improves the overall performance of camera localization systems, enabling more accurate and reliable localization in real-world applications. The motion blur removal technique can assist in overcoming obstacles associated with dynamic environments and moving cameras. The proposed approach represents a substantial development in indoor camera localization by addressing motion blur problems and utilizing a recurrent deep architecture, as shown in Fig. 1. Combining these contributions strengthens the dependability and accuracy of camera localization systems, creating opportunities for further developments in associated research and real-world applications.

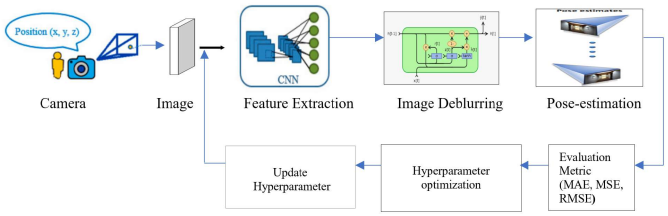

Fig. 1. Overview of proposed recurrent deep architecture

The research contribution is to develop a recurrent deep architecture for improving indoor camera localization integrating with the motion blur elimination process from the indoor images of videos. The more specific research contribution is pointed out:

1) To propose an innovative technique for eliminating motion blur in indoor camera localization. It can significantly minimize motion blur's effect on camera localization's precision by evaluating blur patterns and utilizing cutting-edge image processing techniques.
2) To propose a recurrent deep architecture that increases the accuracy and robustness of the camera pose estimation and takes advantage of the temporal data recorded in the succeeding frames.
3) To integrate a recurrent deep architecture for indoor camera localization with a motion blur removal technique that considerably improves accuracy and robustness.

II. Related Works

This section describes the literature review of motion blur, indoor camera localization, and recurrent neural networks.

A. Motion Blur

In the past decade, blurred images have received considerable attention for camera localization in indoor environments. Camera movements and plane translations cause motion blur. An approach [19] addressed the challenge of evaluating and improving a blurry image by eliminating motion blur. It aims to restore sharpness to a blurry image caused by camera shake or object motion, focused on measuring and minimizing non-uniform motion blur [20]. While its adaptation increases accuracy, sensitivity to noise can limit accuracy. To provide a unique deep learning-based strategy for predicting motion blur accompanied by a data deblurring approach tailored to motion blur. Using CNN's powerful feature learning capabilities [21], CNN accurately predicts complex motion blur [22]. However, real-time processing and handling of complex situations continue to present difficulties. Motion blur is pervasive in indoor camera localization, mainly when using small, transportable devices like cell phones and hidden cameras. Considering a few constraints on a particular type of blur, a strategy [23] has recently been developed to minimize the blur caused by camera object movements. Motion blur [24] in authentic images can be caused by multiple factors involving the camera [25] and object movement, resulting in complicated blur patterns. Uniform deblurring approaches cannot eliminate non-uniform blur [26]. While innovative, there are some issues with dynamic underground environments and accuracy in real-world implementation.

An end-to-end system reconstructing blur-free images [27], it can tolerate only minor Gaussian blur. However, issues with scalability and computational complexity could emerge. A patch-based system [28] to expect frequency information uniform motion blur reduction. Its effectiveness in complex situations with fast speeds still needs to be determined. The most significant research [29] focused on employing an update-level blurred-type classification approach based on a CNN to predict

movement flow from a single blurry image. While it performs satisfactorily in conventional environments, its adaptability and performance in complicated circumstances are challenging. An efficient and flexible deep learning-based technique [30] has been proposed to predict and reduce heterogeneous motion blur. By increasing robustness when addressing externalities, scalability concerns and real-time issues remain.

Motion blur is one of the most noticeable flaws in images taken with handheld cameras [31]. Camera shaking and quick object movements in a dynamic image cause blurred artifacts. A classic coarse-to-fine technique to a CNN [32], a recurrent neural network (RNN) architecture dramatically improves its performance. DeblurGAN [33] is influenced by research on Generative Adversarial Nets (GAN). Add a dark channel to the loss function to reduce pattern artefacts, and lightweight U-nets [34] were used to replace the residual net DeblurGAN. A framework for recovering from a combination of noisy and unclear images, a sharp and clear image [35]. A recurrent network [36] designs that operate on arbitrary duration films. An adaptive temporal blending component on a fast RNN, whereas the information from the previous frame was used by simply copying features. An iterative hidden layer update approach inside a single inter-frame time step ensures that the transmitted hidden state fits the target frame. As a result, effective motion deblurring [37] techniques enhance the dependability of associated industries, such as aerospace, traffic monitoring, army search, satellite, and space imagery. In recent algorithms, deep learning [38] predicts the probability dispersion of motion blur and restores the damaged images [39].

The multi-frame images contain a complex network foundation, and the second uses only a single image [40] to deblur the degraded image. On the other hand, difficulties may arise from computing demands and implementation in complex situations. Deep-learning algorithms are not ideal for single-image deblurring [41] because they require a long time to compute complicated building structures or have particular criteria for blur conditions. Scalability and real-time processing are issues. Non-uniform single-image deblurring or predicting unknown non-uniform blur kernels remains a problematic ill-posed inverse issue for recovering a clear image using a blurred image [42]. Computing complexity and problems in real-time applications can limit performance. The optical flow describes the displacement of nearby frames [43] to assist in learning future neural network models. Complex motion patterns and computational overheads are issues.

B. Indoor Camera Localization

The indoor camera localization measures the camera poses of the query image in a random scene. In indoor camera localization, a single image or image sequence is the input, and the predicted camera poses are the outputs [44]. Real-time applications need to be tested to see if they work correctly, and there are some scalability issues [45]; if these issues are resolved, they can work in different indoor environments. The camera localization [46] problem was first implemented as a localization detection problem [47]. The image was located using an image retrieval system. In diverse environments, robustness may be limited by monocular vision. PoseNet [48] was the first to employ CNNs to predict straight 6-DOF camera pose estimation [49].

Motion blur is the leading cause of performance degradation. Although some images contain texture surfaces free from motion blur, many missing ground-truth scene coordinate labels might cause issues. In Bayesian-PoseNet [50], researchers introduced PoseNet to account for the uncertainty in pose estimation. Localization fails because of constraints, such as motion blur and illumination changes. The localization performance is improved by the deep attention architecture, which reduces the structured dimensionality and addresses challenges such as motion blur and illumination changes. In [51], an hourglass architecture was proposed as the basis for pose regression. This study demonstrates that the method works on data with motion blur and lighting change problems. Camera localization is restricted [52] employing conditional generative adversarial networks and the regression model to achieve pose estimation. Some other research has focused on frameworks to increase the performance of camera localization [53].

A lightweight CNN for real-time camera localization [54]. It shows a significant improvement in remote sensing applications. However, its accuracy can be hampered by the diversity in terrain features. A method [55] that focuses on critical geometric features [56] through multitasking, which uses information from related activities. Its highlight is that temporal consistency improves accuracy, although scalability and real-time implementation are still challenging. In [57], present a technique based on ConvNet that dynamically predicts the real-time camera pose estimation. Its performance in varied lighting and weather conditions makes robustness and reliability indispensable in practical deployments.

A CNN-RNN network with image sequences to enforce temporal smoothness in camera motion [58]. Another approach developed [59], extensive enhancement in video data localization. Still, there are issues with generalizability in other contexts and real-time performance [60]. Utilized deep correlation alignment networks to recognize 3D CAD models [61], [62], a combined dataset of both natural and artificial images for object recognition; and in [63], mapped synthetic to authentic images in benchmark representations to bridge the gap between synthetic and natural images in pattern representations. Additionally, some researchers, such as [64], [65], have synthesized images using 3D models to evaluate search images against a database of synthetic images using in-depth features, specifically for the geolocation image dataset. Inspired by this research, in [66], syntactic and actual image sequences are used as a training

dataset and the PoseNet network to predict the camera pose directly.

A recurrent neural network [67], domain adaptation, addressed the enormous visual and domain-specific differences between artificial and real images, significantly degrading localization accuracy [68]. Indoor camera localization based on deep learning remains challenging because indoor scenes have motion blur. Deep learning-based approaches overcome the limitation of local feature-based methods but are still far from having actual value. Some indoor positioning applications, such as mobile guide robots [69] and mobile robots [70], require more accurate camera positioning. Therefore, there is a need to improve the localization of RNN-based indoor cameras.

C. Recurrent Neural Network

The ability of recurrent neural networks (RNN) to handle sequential data has led to their widespread application in camera localization tasks. LSTM models have been used successfully to determine the exact location inside cameras by taking advantage of their capacity to collect long-range dependencies and handle sequential inputs [71]. LSTM can accurately anticipate the camera's position because it uses the temporal information in the data. Bidirectional Long Short-Term Memory (Bi-LSTM) networks have succeeded in several computer vision tasks, including indoor camera localization. Bi-LSTM models can successfully learn and predict the position of a camera inside an indoor environment by capturing the temporal interdependence and spatial context in sequential data [72]. GRUs have demonstrated potential for modelling sequential data and identifying temporal dependencies, which makes them suitable for evaluating video feeds from indoor cameras. Researchers have used GRUs to achieve precise and reliable localization results, enabling accurate indoor tracking and surveillance.

III. Material and Methods

A novel recurrent deep architecture that minimizes motion blur for indoor camera localization is completely experimental research. Collected data from secondary sources to identify the image features, recognition, recurrent neural network, and optimization techniques. Primary data were gathered from publicly available databases. In addition, the image database can be analyzed using a deep-learning system. Developed a recurrent neural network model to estimate and evaluate the camera pose. The overall camera pose estimation process through the motion blur removal process is shown in Fig. 2.

A. Recurrent Deep Learning Architecture

The process of removing motion blur is discussed in two parts: the blur removal method from images and the blur removal method from video. Both methods are described separately below.

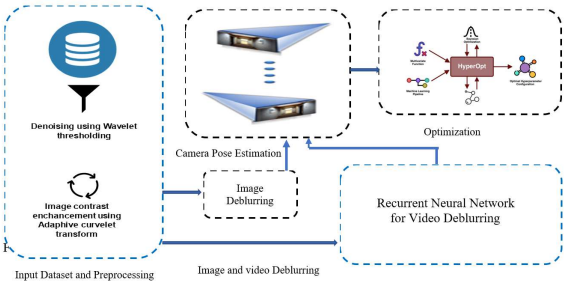

Fig. 2. Camera pose estimation through motion blur elimination process

1) Motion Blur Removal from Single Image: The image deblurring process is shown in Fig. 3. First, blurry images are used as input for a camera pose prediction. In the linear space translation, the blurred image resulting from the relative motion of the camera and the scene can be represented as a two-dimensional convolution model [73]–[77]. A 2D input image is created as:

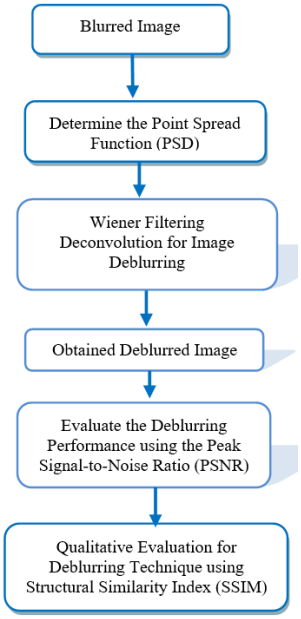

Fig. 3. Image deblurring process

$$b = c * x + \eta \tag{1}$$

Here, b is a 2D blurry image, c is a shift-invariant 2D convolution kernel or point spread function (PSF), x is a 2D clear image, and η is a signal-independent noise term. Equation (1) can be written on the Fourier domain:

$$\tilde{x} = F^{-1}\{F\{b\}.F\{c\}\} \tag{2}$$

Inverting Equation 2 is inverse filtering:

$$\tilde{x} = F^{-1}\frac{F\{b\}}{F\{c\}} \tag{3}$$

Once the motion blurs, apply a deblurring technique, such as Wiener filtering deconvolution, to remove the blur and obtain a clear copy of the original image. A damping factor is added to the inverse filter when Wiener filtering is applied to the deconvolution problem.

$$\tilde{x} = F^{-1}\{\frac{|F\{c\}|^2}{|F\{c\}|^2 + \frac{1}{SNR}} . \frac{F\{b\}}{F\{c\})}\} \tag{4}$$

Where the signal-to-noise ratio (SNR) is infinite when there is no noise in the measurements, Wiener filtering is the same as inverse filtering in that specific scenario. Equation (4) adds a per-frequency damping factor in all other instances, provided that the noise power spectral density and signal magnitude are known for each frequency.

Additionally, the image quality can be enhanced by denoising techniques and improving overall image quality. Even after deblurring, some noise might still be present in the image. Evaluate the performance of the motion blur reduction algorithm by comparing the original and deblurred images or by using the peak signal-to-noise ratio (PSNR). Determine whether the algorithm successfully reduces motion blur through a comprehensive analysis [78], [79]. The PNSR is expressed as:

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}(f(x,y) - \tilde{f}(x,y))^2 \tag{5}$$

Where MSE is the mean square error, f(x,y) is the intensity of the pixel (x,y) before motion deblurring, $\tilde{f}(x,y)$ is the intensity of the pixel (x,y) after motion deblurring, M and N is the size of an image.

$$PSNR = 10\log_{10}(\frac{255^2}{MSE}) \tag{6}$$

Typically, the higher the PSNR, the better the restoration quality; an optimal PSNR is infinity.

Evaluate each technique's deblurring quality using the structural similarity index (SSIM) [80].

$$SSIM(f,\tilde{f}) = \frac{(2\mu_f\mu_{\tilde{f}} + C_1) + (2\sigma_{f\tilde{f}} + C_2)}{(\mu_f^2 + \mu_{\tilde{f}}^2 + C_1)(\sigma_f^2 + \sigma_{\tilde{f}}^2 + C_2)} \tag{7}$$

Where μ_f, $\mu_{\tilde{f}}$, σ_f, $\sigma_{\tilde{f}}$, and $\sigma_{f\tilde{f}}$ is the means, standard deviations, and cross-covariance for the image f and $\tilde{f}$ respectively. C_1 and C_2 are two constants to avoide equations divided by zero. Usually, the higher the SSIM, the larger the restoration quality; an optimal SSIM has a value of 1.

2) Motion Blur Removal from Video: A video deblurring process can be achieved by implementing recurrent neural networks [81]. The Long Short-Term Memory (LSTM) architecture is one of the most popular and ancient architectures for resolving the vanishing gradient problem of RNNs. The LSTM gates decide what information should be forgotten in the hidden states. Each information element is saved in a single state, and the gates are designed to simulate the long-term dependencies in sequential data, which may eventually result in short-term optimization [82]. In terms of improving the short-term memory from an auxiliary module producing extra hidden states, the recurrent neural network design [83] is shown in Fig. 4.

The recurrent network architecture uses a feature generator to generate an updated hidden state h_t and a feature f_t. The feature generator uses the input blurry image B_t and the hidden state h_{t-1} from the previous time step as input.

$$f_t, h_t = Recurrence(B_t h_{t-1}) \tag{8}$$

RNNs use a single hidden state at each time step to store information from the past frames. The hidden state at each step is optimized to maximize the deblurring performance of the corresponding frame [84]. The recurrence module generates an auxiliary state with complementary information from the hidden state.

$$\tilde{h_t} = Recurrence(h_t, \tilde{h}_{t-1}). \tag{9}$$

The outputs $\tilde{h_t}$ is generated by looking into the history and the temporal changes of h_t. The previous input frame is the hidden state h_{t-1} for transmitting information. Through feature combination, B_{t-1} is frequently utilized in conjunction with B_t in video deblurring RNNs [85]. Since the feature f_t in the baseline architecture does not spread to the subsequent frames, it is more particular to the target image at time t. Adaptive blending creates attention maps using the image attributes from both the current and previous frames. Focus is placed on the feature f_t that, at time t, is most relevant to the goal image.

$$\tilde{f}_t = \tilde{w}_{t-1} \times \tilde{f}_{t-1} + w_t \times f_t \tag{10}$$

where $\tilde{w}_{t-1} \geq$ and $w_t \geq 0$.

All features retrieved from the adaptive blending and recurrence modules are combined in the reconstruction module. After combining all features $(f_t, h_t, \tilde{f}_t, \tilde{h}_t)$, many convolutional layers generate the deblurred image.

$$L_t = Reconstructor(f_t, h_t, \tilde{f}_t, \tilde{h}_t) \tag{11}$$

Comparing the output L_t to the ground-truth sharp image S_t trains the recurrent model using a supervised loss function as $\left\|L_t - S_t\right\|$.

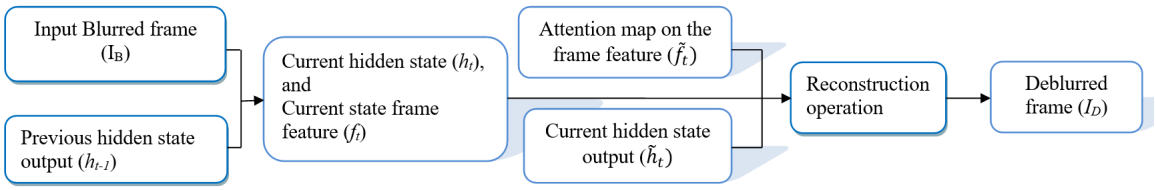

Fig. 4. Video deblurring process

B. Camera Pose Estimation

This section describes a camera localization approach integrating an indoor camera localization system based on a recurrent neural network with a motion blur elimination process. It explains how to effectively remove motion blur and incorporate it into a localization system to improve the camera localization process. Develop a recurrent neural network approach for camera pose estimation. CNN extracts image features from the input image sequence, and LSTM determines pose loss. Refer to one type of RNN as Long-Short-Term Memory (LSTM), which prevents the disappearance of gradients. LSTMs using a technique known as gates can learn long-term dependencies. Several advanced, recurrent architectures, including LSTM and GRU.

Indoor camera localization describes several phases to predict the camera pose from image sequences or videos accurately. First, a short video is provided as input, and the deblurring algorithm analyzes the motion blur elimination process. Subsequently, a recurrent neural network (RNN) predicts the camera pose using information extracted from the input image or video. The evaluation process helps to improve the functionality of the overall camera localization system. A hyperparameter optimization technique was applied to improve system performance and ensure optimal results. By repeatedly updating the hyperparameters, the camera localization process continuously enhances and increases the ability to predict the camera pose more accurately in different environments.

Here, the measurement technique for localization accuracy is discussed, which mainly uses the Euclidean distance to calculate the pose error (X_{cm}, Y°) of the proposed camera localization system. The defined thresholds of the three groups, such as best, average, and worst pose error, are (0.25m, 2°), (0.5m, 5°), and (0.5cm, 10°), respectively. The absolute difference between the predicted position, the orientation value, and the actual position and orientation values measures the accuracy of the pose estimation. The position error is calculated based on the Euclidean distance between the expected and the exact original values of the camera. The value for the position error is:

$$P_error = \left\| C_{est} - C_{gt} \right\|_2 \tag{12}$$

C_est is the estimated value of the camera pose error, and C_gt is the actual camera value of the origin. The orientation error $|\alpha|$ is measured by convention:

$$2cos|\alpha| = trace(R_{gt}^{-1} R_{est}) \tag{13}$$

Here, $|\alpha|$ is the smallest rotation angle required to align the estimated rotation matrix R_{est}t with the ground truth value of the rotation R_{gt}. The deep learning-based pose estimation technique is a classification that only estimates the pose of the images. They represent the camera pose mathematically as:

$$loss(I) = \left\| C_{est} - C_{gt} \right\|_2 + \left\| R_{est} - \frac{R_{gt}}{\left\| R_{gt} \right\|} \right\|_2 \tag{14}$$

Where $[C_{gt}, R_{gt}]$ is the ground truth pose, and β is the hyperparameter that determines the relative weight of the orientation and position errors that depend on the training dataset.

C. Datasets

The availability of public databases for localization activities has recently increased, focusing on utilizing deep-learning approaches for image processing. Only a few public databases exist, such as Microsoft Researchers 7-Scenes [86], and InLoc [87]. Assessing advanced indoor camera localization algorithms for large-scale multidimensional datasets incorporating diverse collection platforms, environments, and images is crucial. The Microsoft 7-Scenes dataset developed by Microsoft Research was utilized in this study. It contains seven indoor environments and is a widely used RGB-D dataset. The images were captured using a handheld Kinect camera with a resolution of 640×480 pixels, and the ground-truth camera positions were obtained through Kinect fusion. A detailed 3D model accompanied each scene, and multiple sequences of tracked RGB-D camera frames were split into training and testing data. The 7-Scene dataset offers a variety of indoor scenes, including offices, chess, heads, stairs, etc., to train localization algorithms. It provides ground truth camera values, which help evaluate the accurate camera localization. This dataset provides colour and depth information of the images; it improves localization accuracy compared to supplying RGB image files only. This dataset is frequently used in indoor localization systems because it combines indoor scenes, ground truth values, colour, and

depth image information values. However, this dataset presents significant challenges, owing to factors such as motion blur, textureless surfaces, and repetitive structures [88]. Fig. 5 shows the input, ground truth, and predicted image.

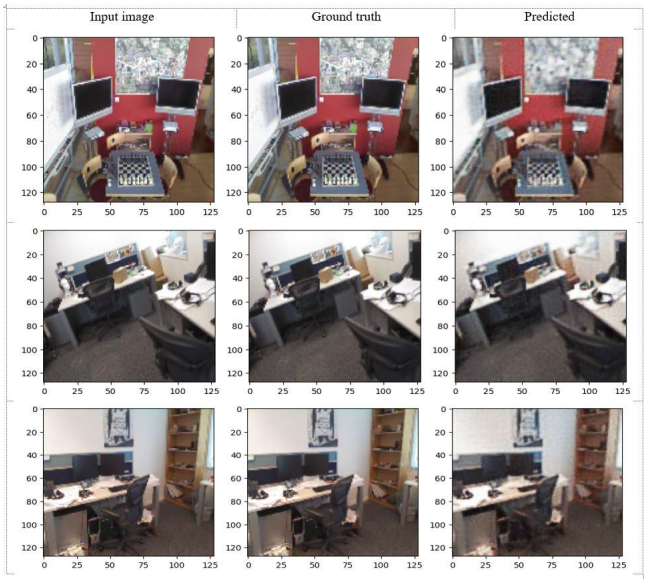

Fig. 5. Input images, ground truth images, and predicted images of 7-scenes dataset.

D. Data Pre-Processing and Optimization

The network is trained uniformly on different datasets, such as 7-Scenes, by resizing the images to 256 pixels. Subsequently, the input images were adjusted to have intensity values between -1 and 1. The ResNet34 component of the network was pre-trained on the ImageNet dataset, whereas the other elements were randomly initialized. We shrank 256×256 pixel images for the network during the training and testing phases by applying an arbitrary and centralized cropping mechanism. The augmentation phase was necessary to increase the generalization capabilities of the architecture under various meteorological scenarios and the duration of the scenarios. Our methods are in Python 3.10 using PyTorch [89] and Adam solvers with a learning rate of 5×10^{-7}. On a GPU, we trained the network with a few hyperparameters, such as epoch is 50, batch size is 64, train dropout is 0.6, test dropout is 0.1, validation frequency is 5, weight decay is 0.0003, learning rate is $5e^{-05}$, weight initialization β is 0.8, and γ is 0.3.

In Adam Solver [90], the objective function to be minimized to measure the camera pose is first defined using the Adam optimization technique. A loss function calculates the difference between the predicted and actual image features. Then, the camera parameters are modified periodically to decrease specific loss functions. It is easy and efficient to optimize the camera pose by computing the gradient of the loss function with the camera pose. Adam optimization updates the parameters to connect the observed image features to the corresponding camera precisely pose estimates.

E. Model Evolution Metrics

Error calculation techniques were applied to evaluate the effectiveness of the recurrent neural network model Mean Absolute Error (MAE), Mean Square Error (MSE), and Root Mean Square Error (RMSE) are most of them.

1) Mean Absolute Error (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid y_i - x_i \mid \tag{15}$$

x_i is the predicted value, and y_i is the mean value.

2) Mean Square Error (MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2 \tag{16}$$

x_i is the predicted value, and y_i is the mean value.

3) Root Mean Square Error (RMSE):

$$RMSE = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2} \tag{17}$$

x_i is the predicted value, and y_i is the mean value.

F. Summary

This section extensively describes a novel approach that combines a recurrent neural network with the motion blur removal procedure. It also includes motion blur elimination methods for single images, image sequences, and videos, as well as camera pose prediction. The dataset set, data preprocessing, and model evaluation processes are also thoroughly described. Our proposed approach outperforms existing research in accuracy and provides a foundation for enhancing the performance of computer vision applications where motion blur is a significant issue.

IV. Results and Discussion

A. Motion Blur Elimination

Several methods, including peak signal-to-noise ratio (PSNR) and SSIM, evaluate how well the motion blur removal process has worked. This research uses PSNR and structural similarity index (SSIM) to assess the motion blur removal process.

1) PSNR: Evaluate the performance of the motion blur elimination algorithm by comparing the original and deblurred images or by using the peak signal-to-noise ratio (PSNR). The higher the PNSR value, the better the motion blur removal process. Fig. 6 shows that the PSNR values are very close among the seven scenes, ranging from 28.23 to 31.96. The lowest PSNR is 28.23 for the pumpkin scene and 28.23 for the

head scene, and the highest PSNR is 31.96 for the head scene. The PSNR values of the remaining five scenes are relatively close: Office 28.56, Redkit 28.65, Stairs 29.37, Chess 29.76, and Back 30.12. Typically, the higher the PSNR, the better the restoration quality. If the PSNR is not optimal at any point, it is considered an improved value. As such, the quality of the motion blur removal process is better for every scene evaluated in this research study.

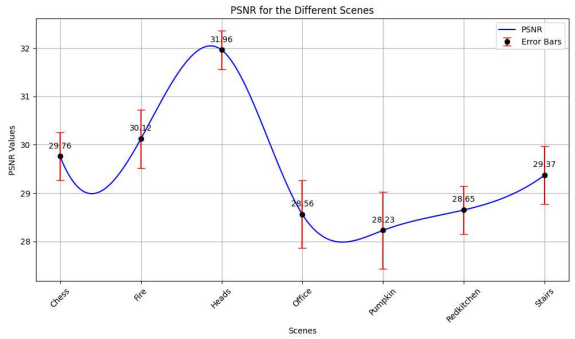

Fig. 6. Peak signal-to-noise ratio (PSNR) of different scenes

2) SSIM: Evaluate the effectiveness of motion blur elimination algorithms using the structural similarity index (SSIM) or comparing the original and blurred images. The motion blur elimination process performs better when the SSIM value is higher. The SSIM values, which range from 0.8572 to 0.9212, are relatively close among the seven cases, as Fig. 7 demonstrates. The SSIM for the chess scenario is 0.9212, whereas the minimum is 0.8572 for the fire scene. The five remaining scenes, Heads 0.8816, Pumpkin 0.9001, Office 0.9123, Red Kitchen 86.07, and Stairs 0.8823, have SSIM values that are reasonably near each other. The optimal SSIM value is 1, and the motion blur removal process for each scene evaluated in this resource study is approximately 0.90, regarded as a decent motion blur removal process. In general, the greater the SSIM, the better the recovery value.

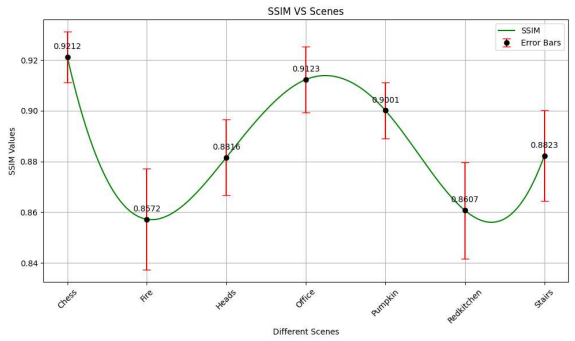

Fig. 7. Structural similarity index (SSIM) of different scenes

B. Camera Pose Prediction

Predict the camera pose once the motion blur has been removed from the image sequence or video. Use seven short video clips comprising 240 frames to improve a camera localization approach. The localization approach estimates the positional and orientational errors. Fig. 8 shows the positional error where the range of positional error is 0.14 to 0.19 meters; the scenes "Head" and "Office" have the least positional error (0.14 meters), while the scene "Stairs" has the most positional error (0.19 meters). The remaining five scenes' location errors are comparatively close to one another. Fig. 9 shows the orientational error, with a range of 3.01° to 9.75°. The orientational error of the "Head" is the worst at 9.75°, while the "Pumpkin" has the lowest at 3.01°. Fig. 8 and 9 demonstrate how pose errors decreased as motion blur was eliminated. Higher accuracy can be attained by lowering the pose loss and removing motion blur from the images.

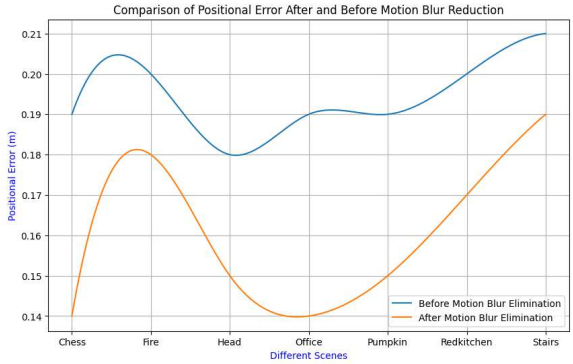

Fig. 8. Positional error for the proposed approach

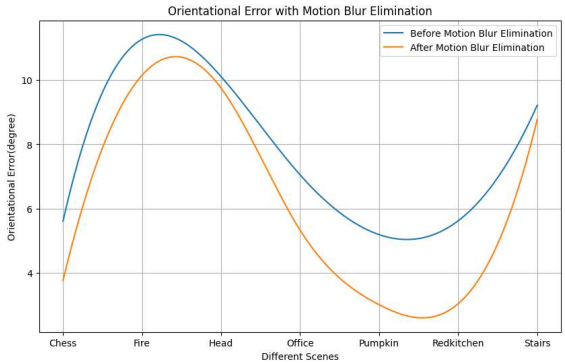

Fig. 9. Orientational error for the proposed approach

C. Evaluation Metrics

The deep learning approach is now trained on each of the seven blurred datasets. After removing motion blur, it calculates each scene's AME, MSE, and RMSE. Next, it calculates the combined distribution of the AME, MSE, RMSE, and standard

deviation for the combined error, and Fig. 10 displays the combined distribution results. It is clear from Fig. 10 how much motion blur affects localization accuracy. According to Table I result, the error rate varies between the scenes for the blurry dataset. With an MAE of 0.0771, an MSE of 0.0055, an RMSE of 0.0923, and a combined error of 0.0379, the ”Fire” scene has the lowest error among the seven scenes. A combined error of 0.0387, an MSE of 0.0087, an RMSE of 0.0975, and an MAE of 0.0831 are the maximum error distribution values for ”Chess.”

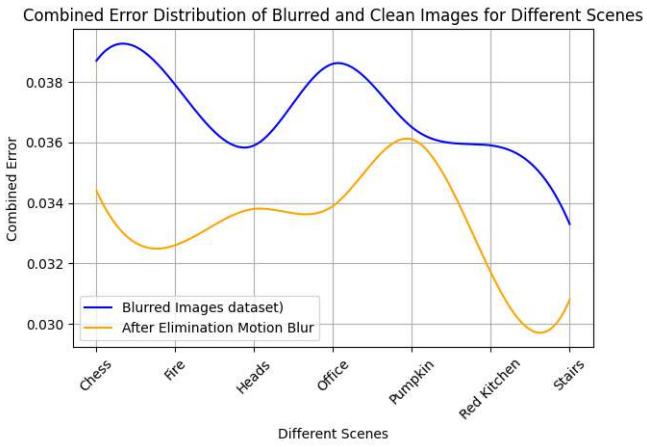

Fig. 10. Compare the combined error distribution between the before and after motion blur elimination

TABLE I. EVALUATION METRIC OF BLURRED IMAGE FOR PROPOSED APPROACH

Scenes	*AME*	*MSE*	*RMSE*	*Combined*
Chess	0.0831	0.0087	0.0975	0.0387
Fire	0.0771	0.0055	0.0923	0.0379
Heads	0.0821	0.0064	0.0992	0.0359
Office	0.0827	0.0056	0.0921	0.0386
Pumpkin	0.0867	0.0069	0.0991	0.0365
RedKitchen	0.0835	0.0067	0.0989	0.0359
Stairs	0.0723	0.0055	0.0949	0.0333

The separate error distributions for the different scenes following the motion blur reduction are shown in Table II. While ”Pumpkin” has a higher error than the other scenes, ”Stairs” has fewer errors. With an MAE error of 0.0643, the ”Stairs” scene has the lowest MAE error, while the ”Pumpkin” scene has the highest MAE error (0.0761). The MSE error for ”Pumpkin” is 0.0059, whereas the MSE error for ”Fire” is 0.0039. 0.0789, the ”Fire” scene has the lowest RMSE error; at 0.0973, the ”RedKitchen” scene has the highest RMSE error. A comparison of the errors for the blurred dataset and after motion blur reduction can be seen in Figure 10, which also shows the overall error distribution for the seven situations. For example, the total error distribution of the blurred dataset in the scene ”Chess” is 0.0387, while the cleaned image dataset has a lower error of 0.0344. The remaining six scenes also demonstrated a discernible difference between the blurred dataset and the after-motion blur reduction. The total error of the after-motion blur reduction is lower than that of the dataset of blurred images, as Fig. 10 shows clearly. The localization performance has significantly improved by removing the motion blur effect from the image or video data.

TABLE II. EVALUATION METRIC OF AFTER MOTION BLUR REDUCTION

Scenes	*AME*	*MSE*	*RMSE*	*Combined*
Chess	0.0711	0.0051	0.0837	0.0344
Fire	0.0652	0.0039	0.0789	0.0326
Heads	0.0712	0.0055	0.0821	0.0338
Office	0.0726	0.0051	0.0809	0.0339
Pumpkin	0.0761	0.0059	0.0876	0.0361
RedKitchen	0.0745	0.0056	0.0973	0.0389
Stairs	0.0643	0.0045	0.0872	0.0348

D. Comparison with Existing Researches

Table III compares the results of the proposed recurrent deep architecture with the most recent research. It displays pose errors for each of the seven scenes, including average and individual, from the proposed study and cutting-edge research using the 7-Scenes dataset. Some recent study findings include PoseNet, MapNet, AtLoc, EpiLoc, and CGAPoseNet. According to Table III, the average positioning error in the current study ranges from 0.18 m to 45 m; PoseNet has the highest average positional error, measuring 0.45 m, while EpiLoc has the lowest average positional error, measuring 0.18 m. Our investigation yielded an average positional inaccuracy of 0.16 m, less than all the studies in Table III. Regarding orientational error, PoseNet has the highest average among current researchers at 9.84° degrees, whereas Deep Attention has the lowest average at 6.25° degrees. Nonetheless, our study yielded an average orientational error of 5.31°, less than EpiLoc’s error. The analysis of the above results shows that our recurrent deep architecture’s results (0.16m, 5.31°) have the lowest pose error when motion blur reduction is employed, which is essential for using more accurate camera localization. The 7-Scenes datasets, used for identifying indoor cameras, contain a lot of intricate issues, like changing viewing angles and lighting conditions. Pose mistakes are higher in image sequences that include more of these issues. Table III shows that compared to other scenes, the orientation error for the ”Head” (9.75°), ”Fire” (10.15°), and ”Stairs” (9.07°) is significantly higher.

E. Discussion

The primary outcome of the research is developing a recurrent neural network to enhance indoor camera localization accuracy by eliminating motion blur issues. We have significantly improved indoor camera localization accuracy by integrating recurrent deep neural network architecture with motion blur removal strategies. In dynamic indoor environments where

TABLE III. AVERAGE POSE ERRORS OF EXISTING ALGORITHMS AND OUR PROPOSED ARCHITECTURE

Network and Ref.	Chess	Fire	Head	Office	Pumpkin	RedKitchen	Stairs	Avg. Pose Error
PoseNet [48]	0.32m,4.06°	0.47m,7.33°	0.29m, 12.00°	0.48m, 6.00°	0.47m, 4.21°	0.59m, 4.32°	0.47m, 06.93°	0.45m, 9.84°
MapNet [91]	0.08m, 3.25°	0.27m, 1169°	0.18m, 13.25°	0.17m, 5.15°	0.22m, 4.02°	0.23m, 4.93°	0.30m, 12.08°	0.21m, 7.78°
AtLoc [92]	0.10m, 4.07°	0.25m, 11.40°	0.16m, 11.80°	0.17m, 5.34°	0.21m, 4.37°	0.23m, 5.42°	0.26m, 10.50°	0.20m, 7.56°
EpiLoc [93]	0.07m, 2.71°	0.24m, 9.18°	0.14m, 12.6°	0.18m, 4.45°	0.18m, 3.32°	0.23m, 4.60°	0.24m, 11.00°	0.18m, 6.82°
CGAPoseNet [94]	0.26m, 6.34°	0.28m, 10.03°	0.17m, 7.98°	0.26m, 7.23°	0.22m, 5.18°	0.55m, 16.7°	0.17m, 12.00°	0.27m, 9.39°
Deep Attention [95]	0.13m, 4.36°	0.22m, 8.04°	0.15m, 8.23°	0.16, 10.54°	0.22m, 4.04°	0.25m, 6.60°	0.21m, 9.36.00°	0.19m, 6.25°
Proposed	0.14m, 3.76°	0.18m,10.15°	0.15m,9.75°	0.14m, 5.33°	0.15m,3.01°	0.17m, 3.04°	0.19m,8.77°	0.16m, 5.31°

motion blur is prevalent, it has been possible to reduce pose error by 20-30% compared to previous research.

The research study's findings have implications for both research and industry. Recurrent neural networks handle temporal dependencies and eliminate motion blur, allowing them to perform very well in sequential data in terms of research. The proposed architecture is essential for several computer vision applications, including guided robots, augmented reality, and indoor robot navigation. Our proposed recurrent neural network improves the accuracy and robustness of indoor camera localization. It is essential for real-world applications like object tracking, indoor mapping, and surveillance that require high security.

The main strength of this research study is to adequately address the core problems of indoor camera localization by combining recurrent deep architecture with motion blur elimination methods. This approach produces very robust results in various adverse environments and real-world situations. Some issues with this research study warrant consideration. Training recurrent deep architectures requires a large amount of labelled data, which is a significant challenge, with changing environments, repetitive data, changing light, and limited viewpoints, among other issues. These shots affect the camera localization results and degrade the accuracy. Since a recurrent neural network can handle the temporal dependency of sequential input, motion blur issues can be handled easily. The light variation, textureless surface, and viewpoint problems are also resolved by syntactically producing the training data using the data augmentation and transfer learning technique. Although this architecture can accurately solve the motion blur problem with synthetic data, its performance may be limited in real-world environments and unpredictable motion blur.

These issues may be addressed in the future through improved data acquisition techniques, more powerful motion blur elimination techniques, and improved model generalization capabilities.

V. CONCLUSION

This article proposes an innovative approach to indoor camera localization that combines motion blur reduction with a recurrent neural network that considers recent advancements in the field. A creative approach that enhances localization accuracy by 20–30% over previous research is used to solve the limitations of the existing research. Although this research substantially improves the accuracy of indoor camera localization, there are certain limitations. The necessity for labelled indoor data for training models is one significant limitation. While data augmentation can address this issue, real-world localization is still constrained to this constraint. In the future, more sophisticated techniques for collecting data in a real-time environment can be explored to solve this problem. The utilization of various sensors, such as LiDAR, WiFi, IMU, and Bluetooth, can provide detailed indoor localization information. Still, the challenge lies in handling the diverse features of each sensor to achieve precise positioning. In future, a more robust and accurate localization system can be developed to address this challenge by effectively combining the data from multiple sensors.

ACKNOWLEDGMENT

The authors wish to thank to Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia.

REFERENCES

[1] M. Sewtz, X. Luo, J. Landgraf, T. Bodenmüller and R. Triebel, "Robust Approaches for Localization on Multi-camera Systems in Dynamic Environments," *2021 7th International Conference on Automation, Robotics and Applications (ICARA)*, pp. 211-215, 2021, doi: 10.1109/ICARA51699.2021.9376475.

[2] M. S. Alam, F. B. Mohamed, A. Selamat and A. B. Hossain, "A Review of Recurrent Neural Network Based Camera Localization for Indoor Environments," in *IEEE Access*, vol. 11, pp. 43985-44009, 2023, doi: 10.1109/ACCESS.2023.3272479.

[3] A. Raza, L. Lolic, S. Akhter and M. Liut, "Comparing and Evaluating Indoor Positioning Techniques," *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1-8, 2021, doi: 10.1109/IPIN51156.2021.9662632.

[4] J. Zhang, and H. Mao, "Wknn indoor positioning method based on spatial feature partition and basketball motion capture," *Alexandria engineering journal*, vol. 61, no. 1, pp. 125–134, 2022, doi: 10.1016/j.aej.2021.04.078.

[5] C. E. A. Bundak, M. A. Abd Rahman, M. K. A. Karim, and N. H. Osman, "Fuzzy rank cluster top k euclidean distance and triangle based algorithm for magnetic field indoor positioning system," *Alexandria Engineering Journal*, vol. 61, no. 5, pp. 3645–3655, 2022, doi: 10.1016/j.aej.2021.08.073.

[6] R. Brylka, U. Schwanecke and B. Bierwirth, "Camera Based Barcode Localization and Decoding in Real-World Applications," *2020 International Conference on Omni-layer Intelligent Systems (COINS)*, pp. 1-8, 2020, doi: 10.1109/COINS49042.2020.9191416.

[7] J. Guo, R. Ni and Y. Zhao, "DeblurSLAM: A Novel Visual SLAM System Robust in Blurring Scene," *2021 IEEE 7th International Conference on Virtual Reality (ICVR)*, pp. 62-68, 2021, doi: 10.1109/ICVR51878.2021.9483818.

[8] H. Yu, H. Zhu, and F. Huang, "Visual simultaneous localization and mapping (SLAM) based on blurred image detection," *Journal of Intelligent & Robotic Systems*, vol. 103, no. 1, 2021, doi: 10.1007/s10846-021-01456-5.

[9] P. Wozniak and B. Kwolek, "Deep Embeddings-based Place Recognition Robust to Motion Blur," *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 1771-1779, 2021, doi: 10.1109/ICCVW54120.2021.00203.

[10] E. Brachmann and C. Rother, "Visual Camera Re-Localization From RGB and RGB-D Images Using DSAC," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5847-5865, 2022, doi: 10.1109/TPAMI.2021.3070754.

[11] H. Yao, R. W. Stidham, Z. Gao, J. Gryak, and K. Najarian, "Motion-based camera localization system in colonoscopy videos," *Medical Image Analysis*, vol. 73, 2021, doi: 10.1016/j.media.2021.102180.

[12] S. Jia, L. Ma, S. Yang and D. Qin, "A Novel Visual Indoor Positioning Method With Efficient Image Deblurring," in *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 3757-3773, 2023, doi: 10.1109/TMC.2022.3143502.

[13] B. Han, Y. Lin, Y. Dong, H. Wang, T. Zhang and C. Liang, "Camera Attributes Control for Visual Odometry With Motion Blur Awareness," in *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 4, pp. 2225-2235, 2023, doi: 10.1109/TMECH.2023.3234316.

[14] H. Li, Z. Zhang, T. Jiang, P. Luo, H. Feng, and Z. Xu, "Real-world deep local motion deblurring," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1314–1322, 2023, doi: 10.48550/arXiv.2204.08179.

[15] P. -E. Sarlin *et al*., "Back to the Feature: Learning Robust Camera Localization from Pixels to Pose," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3246-3256, 2021, doi: 10.1109/CVPR46437.2021.00326.

[16] I. Arrouch, N. S. Ahmad, P. Goh, and J. M. Saleh, "Close proximity time-to-collision prediction for autonomous robot navigation: An exponential gpr approach," *Alexandria Engineering Journal*, vol. 61, no. 12, pp. 11171–11183, 2022, doi: 10.1016/j.aej.2022.04.041.

[17] T. Xie *et al*., "A Deep Feature Aggregation Network for Accurate Indoor Camera Localization," in *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3687-3694, 2022, doi: 10.1109/LRA.2022.3146946.

[18] Q. Li *et al*., "Structure-guided camera localization for indoor environments," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 202, pp. 219–229, 2023, doi: 10.1016/j.isprsjprs.2023.05.034

[19] H. Son, J. Lee, J. Lee, S. Cho, and S. Lee, "Recurrent video deblurring with blur-invariant motion estimation and pixel volumes," *ACM Transactions on Graphics*, vol. 40, no. 5, pp. 1–18, 2021, doi: 10.1145/3453720.

[20] G. Carbajal, P. Vitoria, M. Delbracio, P. Musé, and J. Lezama, "Non-uniform blur kernel estimation via adaptive basis decomposition," *arXiv preprint*, 2021, doi: 10.48550/arXiv.2102.01026.

[21] S. S. Carita, and R. B. Hadiprakso, "Double Face Masks Detection Using Region-Based Convolutional Neural Network," in *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 9, no. 4, pp. 904–911, 2023, doi: 10.26555/jiteki.v9i4.23902.

[22] D. Rozumnyi, M. R. Oswald, V. Ferrari and M. Pollefeys, "Motion-from-Blur: 3D Shape and Motion Estimation of Motion-blurred Objects in Videos," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15969-15978, 2022, doi: 10.1109/CVPR52688.2022.01552.

[23] X. Ge, J. Tan and L. Zhang, "Blind Image Deblurring Using a Non-Linear Channel Prior Based on Dark and Bright Channels," in *IEEE Transactions on Image Processing*, vol. 30, pp. 6970-6984, 2021, doi: 10.1109/TIP.2021.3101154.

[24] J. F. Schmid, S. F. Simon, R. Radhakrishnan, S. Frintrop and R. Mester, "HD Ground - A Database for Ground Texture Based Localization," *2022 International Conference on Robotics and Automation (ICRA)*, pp. 7628-7634, 2022, doi: 10.1109/ICRA46639.2022.9811977.

[25] V. Gampala, M. S. Kumar, C. Sushama, and E. F. I. Raj, "Deep learning based image processing approaches for image deblurring," *Materialstoday:Proceedings*, vol. 10, 2020, doi: 10.1016/j.matpr.2020.11.076.

[26] W. Yang, X. Zhang, H. Ma and G. Zhang, "Laser Beams-Based Localization Methods for Boom-Type Roadheader Using Underground Camera Non-Uniform Blur Model," in *IEEE Access*, vol. 8, pp. 190327-190341, 2020, doi: 10.1109/ACCESS.2020.3032368.

[27] S. Wang, M. Jiu, L. Chen, S. Li, and M. Xu, "A deep encoder-decoder based primal-dual proximal network for image restoration," in *Fifteenth International Conference on Graphics and Image Processing*, pp. 312–322, 2024, doi: 10.1117/12.3021256.

[28] Y. Xu, Y. Zhu, Y. Quan, and H. Ji, "Attentive deep network for blind motion deblurring on dynamic scenes," *Computer Vision and Image Understanding*, vol. 205, 2021, doi: 10.1016/j.cviu.2021.103169.

[29] J. Yu, L. Guo, C. Xiao, and Z. Chang, "Edge-Based Blur Kernel Estimation Using Sparse Representation and Self-similarity," in *Image and Graphics: 11th International Conference*, pp. 179–205, 2021, doi: 10.1007/978-3-030-87358-5_15.

[30] M. Chang, C. Yang, H. Feng, Z. Xu and Q. Li, "Beyond Camera Motion Blur Removing: How to Handle Outliers in Deblurring," in *IEEE Transactions on Computational Imaging*, vol. 7, pp. 463-474, 2021, doi: 10.1109/TCI.2021.3076886.

[31] N. Varghese, A. N. Rajagopalan and Z. A. Ansari, "Real-time Large-motion Deblurring for Gimbal-based imaging systems," in *IEEE Journal of Selected Topics in Signal Processing*, doi: 10.1109/JSTSP.2024.3386056.

[32] C. Zhu *et al*., "Deep recurrent neural network with multi-scale bi-directional propagation for video deblurring," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 3, pp. 3598–3607, 2022, doi: 10.1609/aaai.v36i3.20272.

[33] W. Z. Shao *et al*., "DeblurGAN+: Revisiting blind motion deblurring using conditional adversarial networks," *Signal Processing*, vol. 168, 2020, doi: 10.1016/j.sigpro.2019.107338.

[34] S. Zhang, A. Zhen, and R. L. Stevenson, "Deep motion blur removal using noisy/blurry image pairs," *Journal of Electronic Imaging*, vol. 30, no. 3, pp. 033022–033022, 2021, doi: 10.1117/1.JEI.30.3.033022.

[35] Q. Zhu, M. Zhou, N. Zheng, C. Li, J. Huang and F. Zhao, "Exploring Temporal Frequency Spectrum in Deep Video Deblurring," *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12394-12403, 2023, doi: 10.1109/ICCV51070.2023.01142.

[36] W. Niu, K. Zhang, W. Luo and Y. Zhong, "Blind Motion Deblurring Super-Resolution: When Dynamic Spatio-Temporal Learning Meets Static Image Understanding," in *IEEE Transactions on Image Processing*, vol. 30, pp. 7101-7111, 2021, doi: 10.1109/TIP.2021.3101402.

[37] X. Hu *et al*., "Pyramid Architecture Search for Real-Time Image Deblurring," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4278-4287, 2021, doi: 10.1109/ICCV48922.2021.00426.

[38] M. Tian, Q. Nie and H. Shen, "3D Scene Geometry-Aware Constraint for Camera Localization with Deep Learning," *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4211-4217, 2020, doi: 10.1109/ICRA40945.2020.9196940.

[39] D. Rozumnyi, M. R. Oswald, V. Ferrari, and M. Pollefeys, "Shape from blur: Recovering textured 3d shape and motion of fast moving objects," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29972–29983, 2021, doi: 10.48550/arXiv.2106.0876.

[40] K. Purohit, S. Vasu, M. P. Rao, and A. N. Rajagopalan, "Multiplanar geometry and latent image recovery from a single motion-blurred image," *Machine Vision and Applications*, vol. 33, no. 10, 2022, doi: 10.1007/s00138-021-01254-x.

[41] S. Klenk, L. Koestler, D. Scaramuzza and D. Cremers, "E-NeRF: Neural Radiance Fields From a Moving Event Camera," in *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1587-1594, 2023, doi: 10.1109/LRA.2023.3240646.

[42] D. Park, D. U. Kang, J. Kim, and S. Y. Chun, "Multitemporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training," in *Computer Vision–ECCV 2020: 16th European Conference*, vol. 12351, pp. 327–343, 2020, doi: 10.1007/978-3-030-58539-6_20.

[43] J. Pan, H. Bai and J. Tang, "Cascaded Deep Video Deblurring Using Temporal Sharpness Prior," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3040-3048, 2020, doi: 10.1109/CVPR42600.2020.00311.

[44] T. Xie *et al*., "A Deep Feature Aggregation Network for Accurate Indoor Camera Localization," in *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3687-3694, 2022, doi: 10.1109/LRA.2022.3146946.

[45] R. M. Yuliza, M. Rhozaly, M. Y. Lenni, and G. E. Yehezkiel, "Fast Human Recognition System on Real-Time Camera," *Jurnal Ilmiah Teknik Elektro*

Komputer dan Informatika (JITEKI), vol. 9, no. 4, pp. 895–903, 2023, doi: 10.26555/jiteki.v9i4.27009.

[46] J. Yu *et al*., "CNN-based Monocular Decentralized SLAM on embedded FPGA," *2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pp. 66-73, 2020, doi: 10.1109/IPDPSW50202.2020.00019.

[47] S. Majchrowska et al., "Deep learning-based waste detection in natural and urban environments," *Waste Management*, vol. 138, pp. 274–284, 2022, doi: 10.1016/j.wasman.2021.12.001.

[48] A. Kendall, M. Grimes and R. Cipolla, "PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization," *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2938-2946, 2015, doi: 10.1109/ICCV.2015.336.

[49] Z. Xiao, C. Chen, S. Yang, and W. Wei, "EffLoc: Lightweight Vision Transformer for Efficient 6-DOF Camera Relocalization," *arXiv preprint*, 2024, doi: 10.48550/arXiv.2402.13537.

[50] M. Bui *et al*., "6d camera relocalization in ambiguous scenes via continuous multimodal inference," in *Computer Vision–ECCV 2020: 16th European Conference*, vol. 12363, pp. 139–157, 2020, doi: 10.1007/978-3-030-58523-5_9.

[51] Y. Deng, S. Hui, R. Meng, S. Zhou, and J. Wang, "Hourglass attention network for image inpainting," in *European conference on computer vision*, pp. 483–501, 2022, doi: 10.1007/978-3-031-19797-0_28.

[52] K. Liu, Q. Li, G. Qiu, "Posegan: A pose-to-image translation framework for camera localization," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 166, pp. 308–315, 2020, doi: 10.1016/j.isprsjprs.2020.06.010.

[53] F. Ott, T. Feigl, C. Löffler and C. Mutschler, "ViPR: Visual-Odometry-aided Pose Regression for 6DoF Camera Localization," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 187-198, 2020, doi: 10.1109/CVPRW50498.2020.00029.

[54] A. A. C. Ponce and J. M. Carranza, "Convolutional neural networks for geo-localisation with a single aerial image," *Journal of Real-Time Image Processing*, vol. 19, no. 3, pp. 565–575, 2022, doi: 10.1007/s11554-022-01207-1.

[55] C. Wang *et al*., "DymSLAM: 4D Dynamic Scene Reconstruction Based on Geometrical Motion Segmentation," in *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 550-557, 2021, doi: 10.1109/LRA.2020.3045647.

[56] Y. Cho, S. Eum, J. Im, Z. Ali, H. -G. Choo and U. Park, "Deep Photo-Geometric Loss for Relative Camera Pose Estimation," in *IEEE Access*, vol. 11, pp. 130319-130328, 2023, doi: 10.1109/ACCESS.2023.3325661.

[57] M. Li, J. Qin, D. Li, R. Chen, X. Liao, and B. Guo, "Vnlstmposenet: A novel deep convnet for real-time 6-dof camera relocalization in urban streets," *Geo-Spatial Information Science*, vol. 24, no. 3, pp. 422–437, 2021, doi: 10.1080/10095020.2021.1960779.

[58] R. Clark, S. Wang, A. Markham, N. Trigoni and H. Wen, "VidLoc: A Deep Spatio-Temporal Model for 6-DoF Video-Clip Relocalization," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2652-2660, 2017, doi: 10.1109/CVPR.2017.284.

[59] F. Xue, X. Wu, S. Cai and J. Wang, "Learning Multi-View Camera Relocalization With Graph Neural Networks," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11372-11381, 2020, doi: 10.1109/CVPR42600.2020.01139.

[60] X. Huang *et al*., "Realtime grasping strategies using event camera," *Journal of Intelligent Manufacturing*, vol. 33, no. 2, pp. 593–615, 2022, doi: 10.1007/s10845-021-01887-9.

[61] J. Xiao, L. Li, C. Wang, Z. -J. Zha and Q. Huang, "Few Shot Generative Model Adaption via Relaxed Spatial Structural Alignment," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11194-11203, 2022, doi: 10.1109/CVPR52688.2022.01092.

[62] J. Kim, D. Kim, S. Lee, and S. Chi, "Hybrid DNN training using both synthetic and real construction images to overcome training data shortage," *Automation in Construction*, vol. 149, 2023, doi: 10.1016/j.autcon.2023.104771.

[63] Y. Cai, L. Ge, J. Cai, N. M. Thalmann and J. Yuan, "3D Hand Pose Estimation Using Synthetic Data and Weakly Labeled RGB Images," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3739-3753, 2021, doi: 10.1109/TPAMI.2020.2993627.

[64] Y. Wang, B. Xiao, A. Bouferguene, M. Al-Hussein, and H. Li, "Content-Based Image Retrieval for Construction Site Images: Leveraging Deep Learning–Based Object Detection," *Journal of Computing in Civil Engineering*, vol. 37, no. 6, 2023, doi: 10.1061/JCCEE5.CPENG-5473.

[65] M. Lyu, X. Guo, K. Zhang, and L. Zhang, "A Visual Indoor Localization Method Based on Efficient Image Retrieval," *Journal of Computer and Communications*, vol. 12, no. 2, pp. 47–66, 2024. doi: 10.4236/jcc.2024.122004.

[66] D. Acharya, C. J. Tatli, and K. Khoshelham, "Synthetic-real image domain adaptation for indoor camera pose regression using a 3D model," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 202, pp. 405–421, 2023, doi: 10.1016/j.isprsjprs.2023.06.013.

[67] D. Acharya, S. Singha Roy, K. Khoshelham, S. Winter, "A recurrent deep network for estimating the pose of real indoor images from synthetic image sequences," *Sensors*, vol. 20, no. 19, 2020, doi: 10.3390/s20195492.

[68] N. Li and H. Ai, "Efiloc: large-scale visual indoor localization with efficient correlation between sparse features and 3d points," *The Visual Computer*, vol. 38, pp. 2091–2106, 2022, doi: 10.21123/bsj.2024.9648.

[69] M. S. Alam, F. B. Mohamed, and A. K. M. B. Hossain, "Self-Localization of Guide Robots Through Image Classification," *Baghdad Science Journal*, vol. 21, no. 2(SI), 2024, https://doi.org/10.21123/bsj.2024.9648.

[70] Fahmizal *et al*., "Path Planning for Mobile Robots on Dynamic Environmental Obstacles Using PSO Optimization," in *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 10, no. 1, pp. 166-172, 2024, doi: 10.26555/jiteki.v10i1.28513.

[71] Y. Jin, L. Yu, G. Li, and S. Fei, "A 6-DOFs event-based camera relocalization system by CNN-LSTM and image denoising," *Expert Systems with Applications*, vol. 170, 2021, doi: 10.1016/j.eswa.2020.114535.

[72] M. S. Alam, A. K. M. B. Hossain, and F. B. Mohamed, "Performance Evaluation of Recurrent Neural Networks Applied to Indoor Camera Localization," *International Journal of Emerging Technology and Advanced Engineering*, vol. 12, no. 8, 2022, doi: 10.46338/ijetae0822_15.

[73] H. Yang, X. Su, S. Chen, W. Zhu, and C. Ju, "Efficient learning-based blur removal method based on sparse optimization for image restoration," *PLoS One*, vol. 15, no. 3, 2020, doi: 10.1371/journal.pone.0230619.

[74] J. Dong, S. Roth, and B. Schiele, "Deep wiener deconvolution: Wiener meets deep learning for image deblurring," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1048–1059, 2020, doi: 10.48550/arXiv.2103.09962.

[75] K. -H. Liu, C. -H. Yeh, J. -W. Chung and C. -Y. Chang, "A Motion Deblur Method Based on Multi-Scale High Frequency Residual Image Learning," in *IEEE Access*, vol. 8, pp. 66025-66036, 2020, doi: 10.1109/ACCESS.2020.2985220.

[76] W. Zhou *et al*., "Improved estimation of motion blur parameters for restoration from a single image," *PLoS One*, vol. 15, no. 9, 2020, doi: 10.1371/journal.pone.0238259.

[77] J. S. Oh, H. Lee, and W. Hwang, "Motion blur treatment utilizing deep learning for time-resolved particle image velocimetry," *Experiments in Fluids*, vol. 62, no. 234, pp. 1–16, 2021, doi: 10.1007/s00348-021-03330-4.

[78] Y. Xiang, H. Zhou, C. Li, F. Sun, Z. Li, and Y. Xie, "Application of Deep Learning in Blind Motion Deblurring: Current Status and Future Prospects," *arXiv preprint*, pp. 1–29, 2024, doi: 10.48550/arXiv.2401.05055.

[79] K. -H. Liu, C. -H. Yeh, J. -W. Chung and C. -Y. Chang, "A Motion Deblur Method Based on Multi-Scale High Frequency Residual Image Learning," in *IEEE Access*, vol. 8, pp. 66025-66036, 2020, doi: 10.1109/ACCESS.2020.2985220.

[80] Y. Huihui, L. Daoliang, and C. Yingyi, "A state-of-the-art review of image motion deblurring techniques in precision agriculture," *Heliyon*, vol. 9, no. 6, 2023, doi: 10.1016/j.heliyon.2023.e17332.

[81] J. Park, S. Nah, and K. M. Lee, "Recurrence-in-recurrence networks for video deblurring," *arXiv preprint*, pp. 1–12, 2022, doi: 10.48550/arXiv.2203.06418.

[82] J. Zhao *et al*., "Do RNN and LSTM have long memory?," in *International Conference on Machine Learning*, pp. 11365–11375, 2020, doi: 10.48550/arXiv.2006.03860.

[83] J. W. Rae, A. Potapenko, S. M. Jayakumar, and T. P. Lillicrap, "Compressive transformers for long-range sequence modelling," *arXiv preprint*, pp. 1–19, 2019, doi: 10.48550/arXiv.1911.05507.

[84] J. Dong, S. Roth, and B. Schiele, "Deep wiener deconvolution: Wiener meets deep learning for image deblurring," *Advances in Neural In-*

formation Processing Systems, vol. 33, pp. 1048–1059, 2020, doi: 10.48550/arXiv.2103.09962.

[85] Z. Zhong, Y. Gao, Y. Zheng, and B. Zheng, "Efficient spatio-temporal recurrent neural network for video deblurring," in *Computer Vision–ECCV 2020: 16th European Conference*, pp. 191–207, 2020, doi: 10.1007/978-3-030-58539-6_12.

[86] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi and A. Fitzgibbon, "Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2930-2937, 2023, doi: 10.1109/CVPR.2013.377.

[87] M. Dubenova, A. Zderadickova, O. Kafka, T. Pajdla, and M. Polic, "D-InLoc++: Indoor Localization in Dynamic Environments," *Pattern Recognition*, pp. 246–261, 2022, doi: 10.1007/978-3-031-16788-1_16.

[88] N. Radwan, A. Valada and W. Burgard, "VLocNet++: Deep Multitask Learning for Semantic Visual Localization and Odometry," in *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4407-4414, 2018, doi: 10.1109/LRA.2018.2869640.

[89] S. Imambi, K. B. Prakash, and G. R. Kanagachidambaresan, "PyTorch," *Programming with TensorFlow: Solution for Edge Computing Applications*, pp. 87–104, 2021, doi: 10.1609/aaai.v34i06.6608.

[90] D. Yi, J. Ahn, and S. Ji, "An effective optimization method for machine learning based on ADAM," *Applied Sciences*, vol. 10, no. 3, 2020, doi: 10.3390/app10031073.

[91] A. Hagemann, M. Knorr and C. Stiller, "Deep geometry-aware camera self-calibration from video," *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3415-3425, 2023, doi: 10.1109/ICCV51070.2023.00318.

[92] B. Wang, C. Chen, C. X. Lu, P. Zhao, N. Trigoni, A. Markham, "Atloc: Attention guided camera localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 6, pp. 10393–10401, 2020, doi: 10.1609/aaai.v34i06.6608.

[93] L. Xu, T. Guan, Y. Luo, Y. Wang, Z. Chen, and W. Liu, "Epi-Loc: Deep Camera Localization Under Epipolar Constraint," *Transactions on Internet & Information Systems*, vol. 16, no. 6, 2022, doi: 10.3837/tiis.2022.06.014.

[94] A. Pepe and J. Lasenby, "Cga-posenet: Camera pose regression via a 1d-up approach to conformal geometric algebra," *arXiv preprint*, pp. 1–13, 2023, doi: 10.48550/arXiv.2302.05211.

[95] A. Abozeid, A. I. Taloba, R. M. Abd El-Aziz, A. F. Alwaghid, M. Salem, and A. Elhadad, "An Efficient Indoor Localization Based on Deep Attention Learning Model.," *Computer Systems Science and Engineering*, vol. 46, no. 2, pp. 2637–2650, 2023, doi: 10.32604/csse.2023.037761.