Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.048850

ARTICLE

Efficient Unsupervised Image Stitching Using Attention Mechanism with Deep Homography Estimation

Chunbin Qin* **and Xiaotian Ran**

School of Artificial Intelligence, Henan University, Zhengzhou, 450000, China

*Corresponding Author: Chunbin Qin. Email: qcb@henu.edu.cn

Received: 20 December 2023 Accepted: 13 March 2024 Published: 25 April 2024

ABSTRACT

Traditional feature-based image stitching techniques often encounter obstacles when dealing with images lacking unique attributes or suffering from quality degradation. The scarcity of annotated datasets in real-life scenes severely undermines the reliability of supervised learning methods in image stitching. Furthermore, existing deep learning architectures designed for image stitching are often too bulky to be deployed on mobile and peripheral computing devices. To address these challenges, this study proposes a novel unsupervised image stitching method based on the YOLOv8 (You Only Look Once version 8) framework that introduces deep homography networks and attention mechanisms. The methodology is partitioned into three distinct stages. The initial stage combines the attention mechanism with a pooling pyramid model to enhance the detection and recognition of compact objects in images, the task of the deep homography networks module is to estimate the global homography of the input images considering multiple viewpoints. The second stage involves preliminary stitching of the masks generated in the initial stage and further enhancement through weighted computation to eliminate common stitching artifacts. The final stage is characterized by adaptive reconstruction and careful refinement of the initial stitching results. Comprehensive experiments across multiple datasets are executed to meticulously assess the proposed model. Our method's Peak Signal-to-Noise Ratio (PSNR) and Structure Similarity Index Measure (SSIM) improved by 10.6% and 6%. These experimental results confirm the efficacy and utility of the presented model in this paper.

KEYWORDS

Unsupervised image stitching; deep homography estimation; YOLOv8; attention mechanism

1 Introduction

In the AutoStitch approach proposed by Tian et al. [1], key point recognition and matching were utilized to establish correspondences between two images. Although this method marked a significant advancement in image stitching technology, it was noteworthy that it could lead to ghosting artifacts in the final stitched images.

Convolutional Neural Networks (CNNs) excel in feature extraction, applied in diverse fields like depth estimation, optical flow estimation [1], and deformation correction. In image stitching, researchers explored CNNs and Fully Convolutional Networks (FCNs). Hoang and Tran focused on feature extraction [2,3]. An improved underwater terrain stitching algorithm using spatial gradient

feature blocks was proposed [4]. Evolutionary computing aided in deep learning model design [5]. Nie's view-agnostic approach employed depth homography modules, spatial transformers, and depth image refinement [6,7]. Nie's unsupervised framework addressed baseline changes and pixel-level misalignment in two stages [8]. However, the substantial dimensions of the model posed practical challenges for deployment in real-world scenarios. The extensive size of the model trained through this methodology proved to be impractical for various applications, resulting in suboptimal performance in practical settings. The pixel-wise image stitching network proposed in [9] adopted a large-scale feature extractor and an attention guidance module to obtain high resolution and accurate pixel-level offsets. However, splicing in pixel-level end-to-end networks was not implemented.

To address the issue of the unwieldy size of existing unsupervised image stitching models, we propose a novel unsupervised image stitching method based on the YOLOv8 [10] framework, incorporating a deep homography network and attention mechanism. Recent studies had demonstrated the versatility of the YOLO series algorithms in various detection scenarios [11]. In remote sensing, it had been used for object detection and classification in satellite and aerial images, aiding in land use mapping, urban planning and environmental monitoring [12]. Aboah developed a real-time helmet detection model using YOLOv8 [10], improving both accuracy and speed. A detection function that automatically identifies regions of interest (ROI) has been added, while unnecessary objects were effectively removed [13]. This study aims to augment image stitching through the integration of YOLOv8 edge detection and a deep homography network. The deep homography module handles rotations, scaling, and translations effectively. We introduce an attention mechanism with a pooling pyramid module for improved small object detection. Through meticulous consideration of the vanishing gradient problem, our approach reconceptualizes the calculation of the loss function, thereby enhancing the accuracy of image detection. This method enables precise mask generation for image reconstruction and cropping in stitching scenarios, with the added benefit of reducing the model size. The main contributions of this paper are summarized as follows:

- We design a new unsupervised image stitching method based on the YOLOv8 framework, integrating the deep homography network into the YOLOv8 framework. This method is able to naturally distort the target image to align with the reference image, thereby exhibiting robustness against image distortion problems.
- The attention mechanism is introduced into the YOLOv8 framework, wherein the deep homography network is integrated, and the attention mechanism is combined with the pooling pyramid model to capture multi-scale information while retaining spatial details.
- The calculation of the loss function and the differentiation between images are improved to be more suitable for image stitching tasks.

2 Related Work

This section reviews developments in image stitching and deep homography estimation, focusing on two primary feature-based techniques.

2.1 Feature-Based Image Stitching

Adaptive Warping, such as Dual-Homography Warping (DHW) [14], was designed to address parallax issues by aligning different scene planes independently. While these methods were effective in

simpler scenes, their performance tended to diminish in more complex environments. Techniques like Smoothly Varying Affine (SVA) transformations and As-Projective-As-Possible (APAP) warping, as described by Zaragoza et al. [15], enhanced local alignment by assigning homography to image grids. However, APAP encountered difficulties proximate to object boundaries, particularly in regions with depth variations [15].

Seam-Driven Methods were focused on minimizing stitching artifacts [16]. Seam-based alignment was combined with global homography by Zhang et al. [17], a method also applicable to stereoscopic stitching as explored in [18]. Furthermore, iterative warp and seam estimation were implemented by Lin et al. [19] to identify optimal stitching areas while preserving key structural elements such as curves and lines. While these methodologies generally exhibited effectiveness, their efficacy was strongly contingent on the accuracy of feature detection and may have been constrained in scenarios with sparse features or low resolution.

2.2 Deep Homography Schemes

In the development of deep homography estimation, the initial approach described by Detone utilized a VGG-style network to predict homography by determining offset for the four vertices of an image [20,21]. Expanding upon this, an unsupervised approach with a comparable architecture was introduced in [22], incorporating a novel unsupervised loss function.

In contrast, deep stitching schemes, which were not reliant on the continuous design of geometric features, automatically extracting high-level semantic features from extensive datasets. These methods operated in supervised [19,23–26], weakly-supervised [27], or unsupervised [18] modes, making them adaptable to various complex scenes. However, it faced challenges in handling large parallax, limited by the homography-based alignment model, which could lead to undesirable blurring in areas of parallax during reconstruction. In [28], an adaptive selection algorithm was proposed that sequentially performed fast feature extraction and feature matching, a local deformation method was introduced to smoothly transition overlapping areas, achieving high-precision image alignment. The first unsupervised image stitching framework, introduced in [8], features the design of an unsupervised image reconstruction network aimed at eliminating artifacts from features to pixels. To better realize the stitching of underwater terrain images and solve the problems of slow traditional image stitching speed, reference [4] proposed an improved algorithm for underwater terrain image stitching based on spatial gradient feature block.

3 Improved Image Stitching Model

The section constructs an image stitching method based on the YOLOv8 framework. The methodology is depicted in Fig. 1. Initially, the YOLOv8 framework is customized for image stitching tasks by integrating a depth homography estimation network. This addition complements the existing object detection network and instance segmentation model within the framework. Subsequently, to fully leverage and reuse multi-scale features and enhance small object detection, we introduce the Squeeze-and-Excitation (SE) attention mechanism to address these challenges.

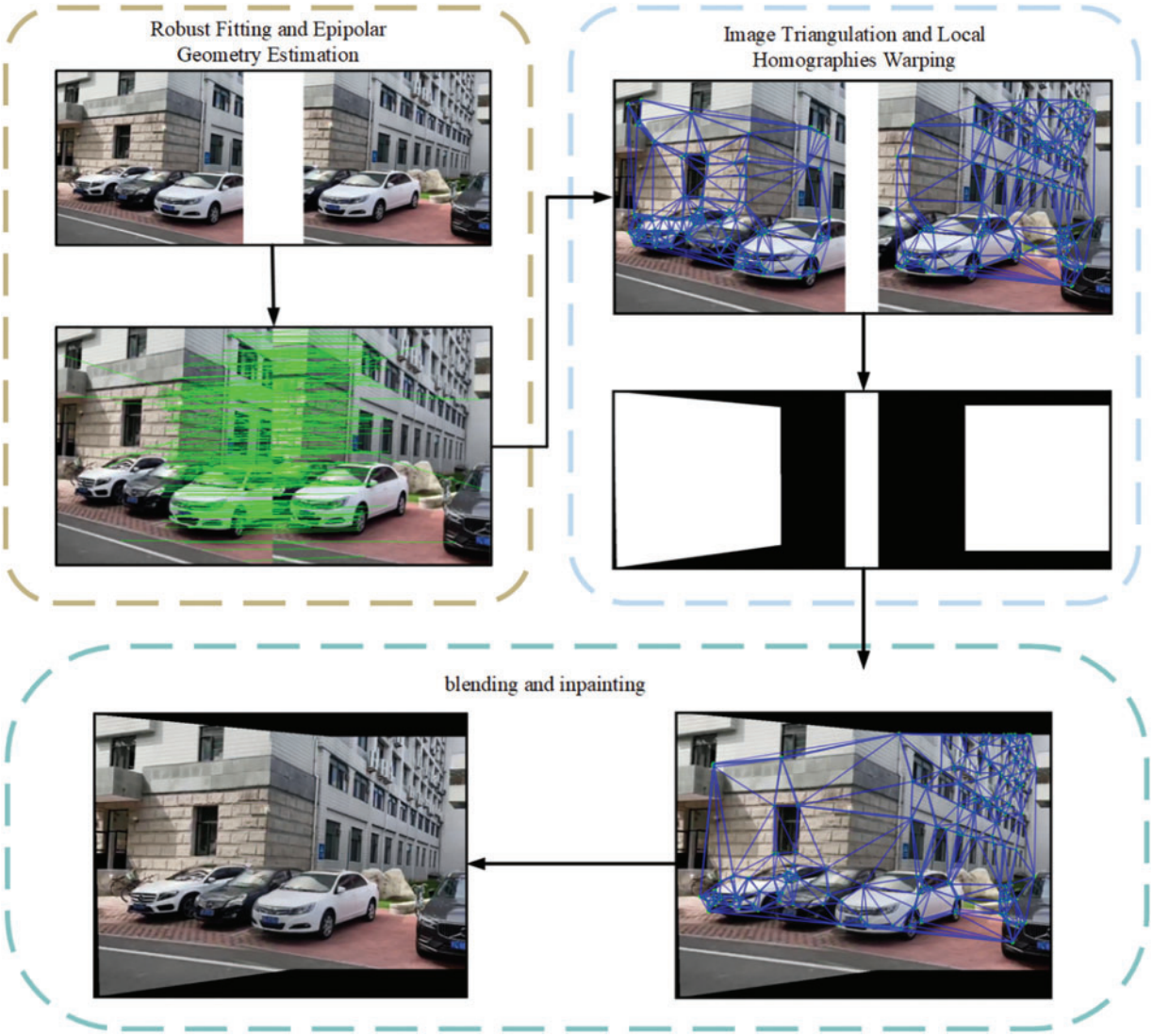

Figure 1: Our image stitching algorithm is divided into three stages, namely the alignment stage, the fusion stage, and the pruning stage. The YOLOv8 algorithm needs to be used in the first and second stages. The YOLOv8 algorithm's network architecture comprises three main components: Input, Backbone, and Head. In the alignment stage, a regression network, which is composed of both convolutional layers and fully connected layers, forecasts offsets, generates mask masks, and learns image structures to train the fusion network for structure splicing; in the construction stage, the spliced structures are the application completes content modification on real data sets

3.1 Improvement of the Backbone

YOLOv8 introduced a distinctive feature with the separation of the detection head into classification and detection heads [29]. For classification loss, it used Binary Cross-Entropy (BCE), while regression loss utilized Generalized Intersection over Union (CIoU) supplemented by Distribution Focal Loss (DFL). The model enhanced small object detection and overlapping area identification by

incorporating an attention mechanism in the backbone network. The amalgamation of upsampling steps prioritized the accentuation of low-resolution target features. Employing triangular grids effectively addresses nonlinear distortion. The triangular grids can flexibly adapt to irregular shapes in the image and achieve smooth transitions to distortion. In computer graphics and computer image processing, triangular mesh processing and rendering algorithms have been widely optimized and applied. Thin-plate spines may require more complex mathematical operations, and their computational costs may be higher. By defining triangles on the image and matching and transforming these triangles during the stitching process, the error during stitching can be reduced. The triangular mesh can help adjust the transformation within each small area, thereby improving the overall stitching accuracy. Additionally, VariFocal Loss (VFL) introduces an asymmetric weighted operation to further refine model performance, as shown in Eq. (1):

$$VFL\,(s_i, s_{i+1}) = -\left((y_{i+1} - y)\log(s_i) + (y - y_i)\log(s_{i+1})\right), \tag{1}$$

where s_i and s_{i+1} represent continuous points in pixels, y is a parameter that influences the transition, y_i and y_{i+1} are to be the drift terms associated with states s_i and s_{i+1}. Upon examining the complete network structure of YOLOv8, it is evident that its backbone is very similar to YOLOv5. Inspired by the Cross Stage Partial (CSP) concept, the C3 module was replaced by the C2f module, which integrates the Efficient Layer Aggregation Networks (ELAN) concept from YOLOv7. For the neck part, YOLOv8 still employed the Path Aggregation Network with Feature Pyramid Network (PAN-FPN) [30,31] feature fusion method. Although feature fusion techniques retain both deep and shallow information, they automatically filter out less apparent and smaller-sized information, making it a primary reason for the non-smooth detection of small objects. Addressing the challenges in unsupervised image stitching with YOLOv8, this method assesses, and reuses features of various scales to achieve more complex and lightweight feature fusion without excessively increasing parameters.

Fundamentally, in deep learning, attention mechanisms quickly survey the entire image and quickly extract valuable data from the vast amount of available information. In the SE layer, features extracted from the input image are subjected to average pooling and then processed through two fully connected layers. The second layer mirrors the neuron count of the input feature layer, maintaining the channel count of the image. The application of the sigmoid function limits the output to a range between 0 and 1, which is then used as a weight multiplier for the original image through the channel attention mechanism. The addition of a Spatial Pyramid Pooling-Fast (SPPF) pooling pyramid module in the last layer of the convolutional neural network presents a method for handling multi-scale features. This module executes pooling operations at various scales, capturing multi-scale information while preserving spatial details. The improved image masks generated through this process are illustrated in Fig. 2.

Represent the input feature image as I^A and a target feature image as I^B, suppose their camera matrices are as Eq. (2):

$$\begin{aligned} P &= K\,[I^A|0]\,, \\ P' &= K'\,[I^B|t], \end{aligned} \tag{2}$$

where $K \in R^{H\times W\times C}$ and $K' \in R^{H\times W\times C}$ are two calibration matrices, P and P' represent the output images of I^A and I^B, respectively. R represents the corresponding image, H, W and C represent the three color channels of the picture, and $t \in R^{3\times 3}$ is translation.

Figure 2: Mask renderings

To enhance the robustness of the fitting method and mitigate calculation errors resulting from incorrectly matched feature pairs, it is imperative to implement a rigorous approach that filters out outliers during the feature-matching process. This involves calculating the mapping error for each feature pair (I^A, I^B) and employing a robust fitting method. The mapping error of a feature pair can be defined and calculated as follows Eq. (3):

$$E_i = \| V\left(I^A + \frac{e'}{z\left(I^A\right)} - I^B\right) \|, \tag{3}$$

where E_i is output as the result, $\frac{e'}{z\left(I^A\right)}$ performs triangular transformation on the input image I^A to calculate the difference with I^B, $V(\cdot)$ normalizes the difference, and outputs it as a mask image, e' is the coordinate pole of I^A. A threshold can be set based on statistical considerations or domain-specific knowledge. Feature pairs with mapping errors exceeding this threshold are considered outliers and are excluded from the subsequent fitting process.

3.2 Improvement of the Head

YOLOv8 introduced a novel state-of-the-art (SOTA) model, encompassing object detection networks with P5 640 and P6 1280 resolutions, as well as the instance segmentation model [32]. YOLOv8 adopts scaling factors akin to YOLOv5, facilitating the creation of models of varying scales. However, this model's performance falls short in image stitching tasks. Therefore, we are revamping the header functionality, removing all header structures, and incorporating a core deep homography estimation network, with the Identity module serving as a pass-through layer.

Deep homography networks are often used as part of a viewless image-stitching framework. Nonetheless, deep homography for image stitching presents greater challenges, as the baseline of the input images is typically extensive. A homography network is crucial to free CNNs from complex tasks. At this stage, our objective is to derive the projective transformation between images, which furnishes alignment data for subsequent stitching procedures. This stage is critical for enabling our network to conduct viewless image stitching. Current unsupervised deep homography approaches used image patches as input, as outlined in [22,33]. The objective function for these methods can be represented by the following as Eq. (4):

$$L_{PW} = \| P\left(I^A\right) - P\left(H\left(I^B\right)\right) \|_1, \tag{4}$$

where $P(\cdot)$ represents extracting an image patch from a complete image and $H(\cdot)$ distorting one shape and aligning it with another image. Utilizes extra content around the target patch to fill inactive pixels

in the distorted target patch. This is called the constraint strategy based on padding. This approach performs effectively in homography estimation with small baselines [33] or medium baselines [22], but does not perform well with large baselines. Especially when the baseline difference is too large and there is not much overlapping area before the input image, the homography estimation of these patches will be meaningless.

To mitigate the challenges, we adopt an ablation-based strategy for unsupervised homography estimation. Concretely, we ingest the complete image as our input, ensuring the inclusion of all overlapping regions. As the warped target image converges towards alignment with the reference image, the conventional practice of populating invalid pixels in the warped image is discontinued. Rather, we opt for an ablation approach, wherein the content of the reference image is selectively excised in the areas corresponding to the invalid pixels in the warped target image. Formally, our unsupervised homography objective function is articulated as Eq. (5):

$$L'_{PW} = \| H(N) \cdot I^A - H\left(I^B\right) \|, \tag{5}$$

where $H(N) \cdot I^A$ is multiply pixels between different images and N is an all-one matrix with identical size with I^A and I^B.

In line with other homography estimation techniques, grayscale images are also employed as input to the network. Initially, a feature extractor is employed to learn the features and the dimensionality of the feature map in the blue section. This involves incorporating two convolutions and a pooling layer within each small blue block, followed by using three convolutions and two fully connected layers. A regression network is composed to process and predict offsets. The calculated global homography furnishes comprehensive homography data and can be readily applied in the subsequent mask-stitching phase. However, feature mismatch may lead to large errors or even inaccuracies in homography estimation, leading to the unavoidable occurrence of ghosting artifacts. Therefore, we improve the structure of the framework to improve the above problems. Our improved structure diagram is shown in Fig. 3.

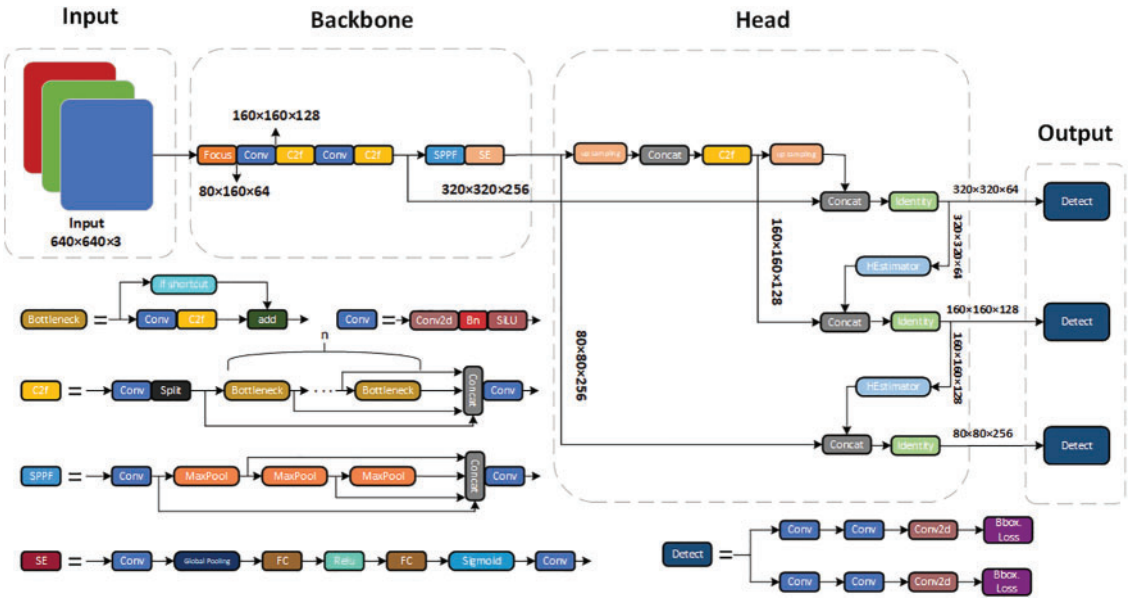

Figure 3: Our-YOLO structure diagram

3.3 The Loss Function

There are permanent problems with backpropagation, such as the vanishing gradient problem. To overcome the vanishing problem, a new anti-vanishing backpropagation learning algorithm called directed random loss descent was introduced in [34]. We redefined the way the loss function is calculated. The loss function calculation in the alignment stage first performs mask processing, thresholds the values, and creates a binary mask, where values greater than one are considered valid pixels. Initialize each element in the loss calculation to a zero tensor and perform different levels of loss calculations. Iterate over the elements and calculate the loss at each level and the mean absolute error with the corresponding region using functions in the framework. Finally, the total loss is computed as the weighted summation of the individual losses at different levels. The formula is shown in Eq. (6):

$$Loss\left(I^A, I^B\right) = \frac{1}{n}\sum_{i=1}^{n} \mid I_i^B - f\left(I_i^A\right) \mid, \tag{6}$$

where the n is the total number of data points, I_i^B is the target image of the *i-th* input, $f\left(I_i^A\right)$ is the predicted value of the *i-th* input data I_i^A by the model, $\mid I_i^B - f\left(I_i^A\right) \mid$ represents the difference between the test and the true value. The differences of the points are summed and averaged to ensure that the loss does not increase infinitely as the number of data points increases.

Seam mask processing is performed on the image and mask generated in the alignment stage, the loss component is calculated from the low resolution and the loss of the original image, and the losses from various components are calculated by hyperparameter weighting. This formula is shown in Eq. (7):

$$Loss\left(I^A, I^B\right) = \frac{1}{n}\sum_{i=1}^{n} \left(I_i^B - f\left(I_i^A\right)\right)^2, \tag{7}$$

when the image is processed into a mask, the n is the total number of data points. $\left(I_i^B - f\left(I_i^A\right)\right)^2$ is to calculate the square of the difference between the predicted and actual values for each data point. Summing the squared differences of all data points and taking the averages.

4 Experiments

This section introduces the data set, experimental platform, evaluation indicators and comparative experimental results used in the experiment. There are performance comparisons of baselines with our model, and ablation experiments. Experimental results verify the effectiveness of the proposed model in target detection tasks.

4.1 Information about Datasets and Dataset-Related Specifics

This study employs the Warped COCO dataset for pre-training. Then it fine-tunes and trains our model on the UDIS [8] dataset (UDIS dataset collected by Nie Lang's team at Beijing Jiaotong University, China) and MVS-Synth dataset [35]. In this part of the dataset, the average overlap rate exceeds 90%, including indoor, outdoor, night, dark, and other variable scenes. Fig. 4 shows some scenes included in the dataset, and Fig. 5 shows the overlap rate of some scene images.

The training set of the UDIS dataset comprises 10,500 samples, while the test set comprises 1100 samples. The training set of the MVS-Synth dataset includes 6200 samples, and the test set includes 150 samples. Due to the presence of numerous small objects or targets in sample images, the samples are resized or standardized to dimensions of 640 × 640 to uphold the accuracy of the entire detection process.

Figure 4: Dataset scenario

Figure 5: Overlap rates

Each stage is refined in comparison to YOLOv8, with training conducted for the warped and fuse networks using the Adam optimizer [36]. The iterative training spans 100 epochs and 50 epochs for the respective networks, incorporating an exponential decay learning rate schedule.

The value is set to 10-4, the first stage batch-size = 340, epoch = 100, the second stage batch-size = 300, epoch = 50. The complete training procedure is unsupervised, signifying that only target images are required as input, not labels. In the test, stitching two input images with a resolution of 640 × 640 took about 0.3 s. To guarantee fairness and comparability when comparing models, no pre-training weights are utilized consistently in all ablation experiments and various model training processes in the comparison experiments.

4.2 Experimental Platform

Regarding hardware and software, we employ an Intel(R) Xeon(R) Platinum 8255C processor with 12 cores and 24 threads, 24 GB of system memory, a GeForce RTX 3090 graphics processor with 24 GB of video memory, and the deep learning model framework PyTorch 1.11.0, YOLOv8's benchmark version and Ultralytic. The system is Ubuntu 18.04.

The experiment utilizes metrics such as PSNR, SSIM and Giga Floating-Point Operations Per Second (GFLOPs) as evaluation indicators. A higher PSNR value indicates a better image quality with less distortion or noise. SSIM values fall on a scale from 0 to 1, and a value closer to 1 suggests that the image is more like the original. The calculation formula of *PSNR* is to evaluate the performance, which can be calculated as Eq. (8):

$$\begin{aligned} MSE &= \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[T\left(i,j\right)-F\left(i,j\right)\right]^2, \\ PSNR\left(I^A,I^B\right) &= 10\cdot\log_{10}\left(\frac{MAX_T^2}{MSE}\right), \end{aligned} \tag{8}$$

where the $T(\cdot)$ is the pixel value of row i and column j in the original image, the $F(\cdot)$ is the value of the corresponding pixel in the image after rotation, m and n represent input image I^A and I^B, respectively. The *MSE* formula calculates the square of the difference between each corresponding pixel, and then substitutes the result into the *PSNR* formula. The MAX_T^2 is the maximum possible pixel value of the image. *MAX* in *PSNR* is the maximum pixel value, which is 255.

The SSIM calculation formula evaluates performance, which can be calculated as Eq. (9):

$$SSIM\left(I^A,I^B\right) = l\left(I^A,I^B\right)^\alpha\cdot c\left(I^A,I^B\right)^\beta\cdot s\left(I^A,I^B\right)^\gamma, \tag{9}$$

where $l\left(I^A,I^B\right)^\alpha$ is used to compare the brightness of two images, $c\left(I^A,I^B\right)^\beta$ is used to compare the contrast of two images, and $s\left(I^A,I^B\right)^\gamma$ is used to compare the structural information of two images. α, β and γ are used to adjust the weight of the comparison function.

The *PSNR* (I^A, I^B) and the *SSIM* (I^A, I^B) contribute to the calculation of *PSNR* and *SSIM* between two images, respectively.

4.3 Comparison Experiment

A series of empirical investigations are undertaken to assess the efficacy of the Our-YOLO methodology. The comparative methodologies encompass global homography (Homo), APAP [15], USIS-RSFI [8], and the benchmark YOLOv8 model [37]. The experimentation involves subjecting all evaluation methodologies to scrutiny across two distinct datasets. Emphasizing precision in image alignment, the resultant stitch outputs are derived through the computation of the mean blend of their respective datasets.

Table 1 presents the tabulated outcomes of diverse methodologies. In specific instances, APAP [15] exhibit suboptimal performance, manifesting in pronounced misalignment and consequential insubstantial indexing, denoted by "−". Homo demonstrates limited efficacy in addressing substantial disparateness and mitigating local structural misalignment, thereby yielding the lowest scores.

Table 1: Comparative test of PSNR and SSIM in each model

Dataset	Indicators	Homo	APAP [15]	UDIS-RSFI [8]	Our model
UDIS [8]	PSNR (↑)	21.25	21.84	23.80	**26.34**
	SSIM (↑)	0.7105	0.6952	0.7929	**0.8414**
MVS-Synth [35]	PSNR (↑)	17.80	21.25	24.56	**26.42**
	SSIM (↑)	0.6308	0.8434	0.8345	**0.8494**

(Continued)

Table 1 (continued)

Dataset	Indicators	Homo	APAP [15]	UDIS-RSFI [8]	Our model
	Model size (MB)	–	–	2105	**188.3**
	GFLOPs	–	–	10.5	**14.5**

Conversely, the three extant methodologies showcase superior alignment quality, resulting in elevated scores. Notably, our proposed methodology consistently attains the highest scores across most test cases, substantiating its superior alignment quality and efficacy. The model's test outcomes are displayed in Table 1.

Fig. 6 illustrates the comparative outcomes of the outdoor test case, characterized by diminutive targets, including intricate textures on floor tiles. Two discernible regions within the overlapping domains of each panorama are delineated through colored boxes and corresponding arrows. Ghosting effects, denoted by red arrows, are evident to varying degrees in the results produced by the four extant methodologies. Both global homography and APAP exhibit challenges in aligning structures such as buildings, while UDIS-RSFI demonstrates a capacity to mitigate structural misalignment, as indicated by the red box. Leveraging the attention mechanism, our local deformation model adeptly identifies and aligns diminutive targets and objects, surpassing all other methodologies' performance.

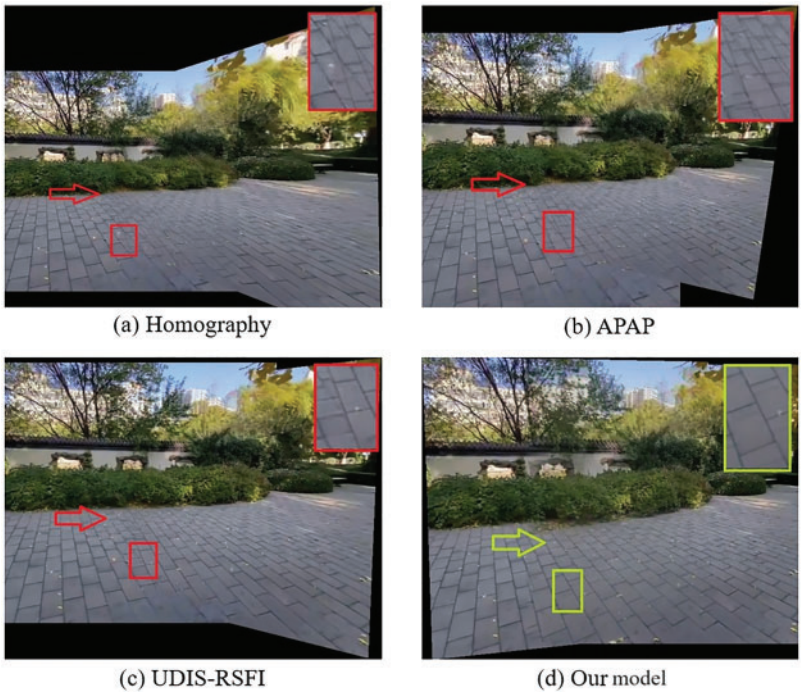

Figure 6: Image stitching quality comparison

4.4 Ablation Experiment

This section presents an ablation experiment conducted to evaluate the performance enhancements of our proposed YOLO model over the baseline models. The modifications include the addition of an Attention Mechanism Module (A) and a Deep Homography Networks Module (B). The performance of each model is assessed using two datasets, UDIS and MVS-Synth, and two indicators, PSNR and SSIM.

The datasets used in our experiments are UDIS and MVS-Synth. The performance indicators include PSNR for image quality assessment and SSIM for perceived image quality.

From Table 2, we can draw the following conclusions, the computational complexity is 14.1 for YOLOv5, 14.3 for YOLOv8, YOLOv8+A, and YOLOv8+B, and marginally higher at 14.5 for Our model. This indicates a slight increase in computational demand for the added modules.

Table 2: Comparative analysis

Dataset	Indicators	YOLOv5	YOLOv8	YOLOv8 + A	YOLOv8 + B	Our model (A + B)
UDIS [8]	PSNR (↑)	22.23	22.42	23.13	25.42	**26.34**
	SSIM (↑)	0.7305	0.7329	0.7529	0.8103	**0.8414**
MVS-Synth [35]	PSNR (↑)	22.12	22.13	23.36	25.11	**26.42**
	SSIM (↑)	0.7408	0.7483	0.7645	0.8173	**0.8494**
	GFLOPs	14.1	14.3	14.3	14.3	**14.5**

- Ablation experiments of our model with the baseline model, adding attention mechanism module (A), deep homography networks module (B), and Our model (A+B).

The ablation study demonstrates that integrating the attention mechanism and deep homography networks modules systematically enhance the model's performance across both datasets and evaluation metrics. Our YOLO model, which incorporates both modules, achieve the highest PSNR and SSIM values. This validates our hypothesis that these modules contribute significantly to the model's effectiveness in object detection tasks.

The results, presented in Fig. 7, showcase the stitching outcomes obtained through the proposed algorithm, utilizing randomly selected images. Incorporating deep homography estimation, we learned content mask rules to enhance the precision of edge region recognition. Utilizing convolutional operations and attention mechanisms, we determined the image edges and overlapping areas. The mask images outputted by the deep homography networks module represent pixel-level reconstructions of the input images, demonstrating the consistency and effectiveness of our approach in image stitching across multiple horizontal and vertical perspectives.

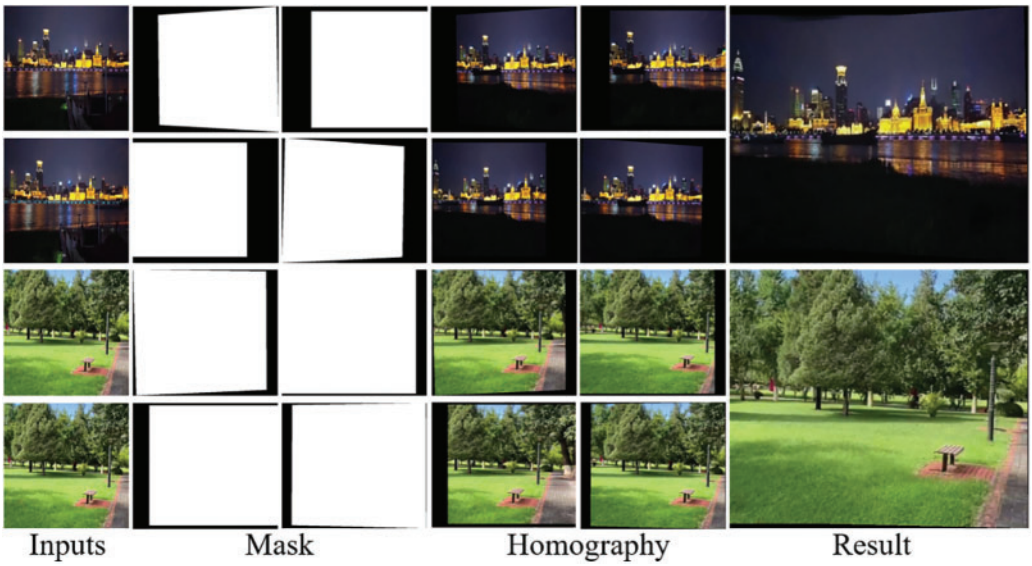

Figure 7: Image stitching quality comparison

5 Conclusion

This study systematically validates the effectiveness of our image stitching methodology, supported by a notable improvement of 10.6% and 6% in PSNR and SSIM metrics on the UDIS dataset, and a significant enhancement of 7.5% and 1.7% on the MVS-Synth dataset. These improvements surpass established methodologies, including global homography, APAP, USIS-RSFI, and YOLOv8. Our approach, characterized by an intricate attention mechanism, adeptly addresses challenges inherent in traditional methods, specifically disparities and local structural misalignments. Notably, it excels in the precise alignment of diminutive targets and intricate details, showcasing superior performance. The method exhibits proficiency in discerning image splicing nuances through content masks and homography estimation, employing convolution operations and attention mechanisms to identify edges and overlapping areas. Pixel-level reconstructions underscore its high coherence, demonstrating effective image stitching across diverse perspectives. Nevertheless, challenges arose during our experiments. Limited image overlap may lead to increased homography errors, heightened splicing artifacts, and potential failures. Subsequent research endeavors will prioritize the mitigation of these challenges and the refinement of specialized detection tasks.

In conclusion, our unsupervised framework and attention to complex image attributes set a new standard in the field, offering a robust, efficient, and versatile solution for computer vision and image processing applications.

Acknowledgement: The authors would like to thank the editors and reviewers.

Funding Statement: This work was supported by Science and Technology Research Project of the Henan Province (222102240014).

Author Contributions: Study conception and design: Chunbin Qin; data collection: Chunbin Qin, Xiaotian Ran; analysis and interpretation of results: Xiaotian Ran; draft manuscript preparation: Xiaotian Ran. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The UDIS dataset used in this article can be obtained from the following link: https://drive.google.com/drive/folders/1kC7KAULd5mZsqaWnY3-rSbQLaZ7LujTY; The MVS-Synth dataset used in this article can be obtained from the following link: https://phuang17.github.io/DeepMVS/mvs-synth.html.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. Tian, Z. Tu, D. Zhang, J. Liu, B. Li and J. Yuan, "Unsupervised learning of optical flow with CNN-based non-local filtering," *IEEE Trans. Image Process.*, vol. 29, no. 8, pp. 8429–8442, Aug. 2020. doi: 10.1109/TIP.2020.3013168.

[2] V. Hoang, D. Tran, N. Nhu, T. Pham, and V. Pham, "Deep feature extraction for panoramic image stitching," in *Proc. Intell. Inf. Database Syst.*, Phuket, Thailand, Mar. 2020, pp. 141–151.

[3] Z. Zhang, C. Xu, J. Yang, J. Gao, and Z. Cui, "Progressive hard-mining network for monocular depth estimation," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3691–3702, Aug. 2018. doi: 10.1109/TIP.2018.2821979.

[4] Z. Wang, J. Li, X. Wang, and X. Niu, "Underwater terrain image stitching based on spatial gradient feature block," *Comput. Mater. Contin.*, vol. 72, no. 2, pp. 421–428, Jan. 2022. doi: 10.32604/cmc.2022.027017.

[5] N. Li, L. Ma, G. Yu, B. Xue, M. Zhang and Y. Jin, "Survey on evolutionary deep learning: Principles, algorithms, applications, and open issues," *ACM Comput. Surv.*, vol. 56, no. 2, pp. 1–34, Jun. 2023. doi: 10.1145/3603704.

[6] L. Nie, C. Lin, K. Liao, M. Liu, and Y. Zhao, "A view-free image stitching network based on global homography," *J. Vis. Commun. Image Represent.*, vol. 10, no. 10, pp. 102950–102962, Apr. 2020. doi: 10.1016/j.jvcir.2020.102950.

[7] L. Nie, C. Lin, K. Liao, and Y. Zhao, "Learning edge-preserved image stitching from large-baseline deep homography," Dec. 2020. doi: 10.48550/arXiv.2012.06194.

[8] L. Nie, C. Lin, K. Liao, S. Liu, and Y. Zhao, "Unsupervised deep image stitching: Reconstructing stitched features to images," *IEEE Trans. Image Process.*, vol. 30, no. 12, pp. 6184–6197, Aug. 2021. doi: 10.1109/TIP.2021.3092828.

[9] Q. Jia, X. Feng, Y. Liu, X. Fan, and L. J. Latecki, "Learning pixel-wise alignment for unsupervised image stitching," in *Proc. ACM Int. Conf. on Multimedia*, Vancouver, BC, Canada, Jun. 2023, pp. 522–530.

[10] A. Aboah, B. Wang, U. Bagci, and Y. Adu, "Real-time multi-class helmet violation detection using few-shot data sampling technique and yolov8," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Vancouver, Canada, Dec. 2023, pp. 5349–5357.

[11] Y. Li, Q. Fan, H. Huang, Z. Han, and Q. Gu, "A modified YOLOv8 detection network for UAV aerial image recognition," *Drones*, vol. 7, no. 5, pp. 304–311, Aug. 2023. doi: 10.3390/drones7050304.

[12] J. Terven, D. M. Córdova-Esparza, and J. A. Romero-González, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Mach. Learn. Knowl. Extraction*, vol. 5, no. 4, pp. 1680–1716, Aug. 2023. doi: 10.3390/make5040083.

[13] A. Vats and D. C. Anastasiu, "Enhancing retail checkout through video inpainting, YOLOv8 detection, and deepsort tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Vancouver, Canada, Dec. 2023, pp. 5529–5536.

[14] J. Gao, S. Kim, and M. Brown, "Constructing image panoramas using dual-homography warping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Colorado Springs, CO, USA, Dec. 2011, pp. 49–56.

[15] J. Zaragoza, T. Chin, M. Brown, and D. Suter, "As-projective-as-possible image stitching with moving DLT," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Dec. 2013, pp. 2339–2346.
[16] A. Agarwala *et al.*, "Interactive digital photomontage," in *Proc. ACM SIGGRAPH*, New York, NY, USA, Jan. 2004, pp. 294–302.
[17] F. Zhang and F. Liu, "Parallax-tolerant image stitching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Dec. 2014, pp. 3262–3269.
[18] F. Zhang and F. Liu, "Casual stereoscopic panorama stitching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Dec. 2015, pp. 2002–2010.
[19] K. Lin, N. Jiang, L. Cheong, M. Do, and J. Lu, "Seagull: Seam-guided local alignment for parallax-tolerant image stitching," in *Proc. Euro. Conf. on Comput. Vis. (ECCV)*, Amsterdam, Netherlands, Oct. 2016, pp. 370–385.
[20] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," Jun. 2016. doi: 10.48550/arXiv.1606.03798.
[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Sep. 2014. doi: 10.48550/arXiv.1409.1556.
[22] T. Nguyen, S. Chen, S. Shivakumar, C. Taylor, and V. Kumar, "Unsupervised deep homography: A fast and robust homography estimation model," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2346–2353, Sep. 2018. doi: 10.1109/LRA.2018.2809549.
[23] J. Gao, Y. Li, T. Chin, and M. Brown, "Seam-driven image stitching," in *Proc. Euro. Conf. on Comput. Vis. (ECCV)*, Florence, Italy, Oct. 2013, pp. 270–283.
[24] F. Liu, M. Gleicher, H. Jin, and A. Agarwala, "Content-preserving warps for 3D video stabilization," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 631–639, Jun. 2023. doi: 10.1145/3596711.
[25] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," *Adv. Neural Inf. Process. Syst.*, vol. 28, no. 5, pp. 2017–2025, Mar. 2015. doi: 10.7551/mitpress/7503.003.0004.
[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Dec. 2016, pp. 770–778.
[27] J. Johnson, A. Alahi, and L. Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Euro. Conf. on Comput. Vis. (ECCV)*, Boston, MA, USA, Jun. 2016, pp. 522–530.
[28] N. T. Pham, S. Park, and C. S. Park, "Fast and efficient method for large-scale aerial image stitching," *IEEE Access*, vol. 9, pp. 127852–127865, Nov. 2021. doi: 10.1109/ACCESS.2021.3111203.
[29] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," Jul. 2021. doi: 10.48550/arXiv.2107.08430.
[30] T. Lin, P. Dollr, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Dec. 2017, pp. 2117–2125.
[31] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Dec. 2018, pp. 8759–8768.
[32] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation," in *Proc. IEEE/CVF Int. Conf. on Comput. Vis.*, Seoul, South Korea, Oct. 2019, pp. 9157–9166.
[33] J. Zhang *et al.*, "Content-aware unsupervised deep homography estimation," in *Proc. Euro. Conf. on Comput. Vis. (ECCV)*, Munich, Germany, Aug. 2020, pp. 653–669.
[34] I. Abuqaddom, B. A. Mahafzah, and H. Faris, "Oriented stochastic loss descent algorithm to train very deep multi-layer neural networks without vanishing gradients," *Knowl. Based Syst.*, vol. 230, pp. 107391–107400, Sep. 2021. doi: 10.1016/j.knosys.2021.107391.
[35] P. Huang, K. Matzen, J. Kopf, N. Ahuja, and J. Huang, "DeepMVS: Learning multi-view stereopsis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Dec. 2018, pp. 2821–2830.

[36] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. doi: 10.48550/arXiv.1412.6980.
[37] G. Jocher, A. Chaurasia, and J. Qiu, *Yolo by Ultralytics*. Accessed: Jan. 10, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics