UniStitch: Unifying Semantic and Geometric Features for Image Stitching

Yuan Mei[1,2], Lang Nie[1,*], Kang Liao[3], Yunqiu Xu[4], Chunyu Lin[5], and Bin Xiao[1]

[1] College of Artificial Intelligence, Chongqing University of Posts and Telecommunications, Chongqing, China
{nielang,xiaobin}@cqupt.edu.cn
[2] Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong, China
yyuan.mei@connect.polyu.hk
[3] School of Computing and Data Science, Nanyang Technological University, Singapore
kang.liao@ntu.edu.sg
[4] College of Computer Science and Technology, Zhejiang University, Zhejiang, China
imyunqiuxu@gmail.com
[5] Institute of Information Science, Beijing Jiaotong University, Beijing, China
cylin@bjtu.edu.cn

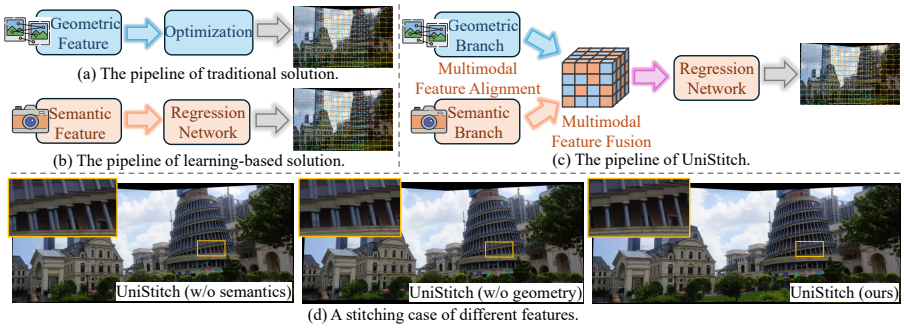

Fig. 1: Existing image stitching solutions vs. UniStitch (ours). The proposed UniStitch integrates the traditional geometric feature and learning-based semantic feature into a unified representation. It effectively eliminates artifacts that persist when either feature modality is used alone, delivering a clear improvement.

Abstract. Traditional image stitching methods estimate warps from hand-crafted geometric features, whereas recent learning-based solutions leverage semantic features from neural networks instead. These two lines of research have largely diverged along separate evolution, with virtually no meaningful convergence to date. In this paper, we take a pioneering step to bridge this gap by unifying semantic and geometric features with **UniStitch**, a unified image stitching framework from multimodal

Work done during the internship at CQUPT. *Corresponding author.

features. To align discrete geometric features (*i.e.*, keypoint) with continuous semantic feature maps, we present a Neural Point Transformer (NPT) module, which transforms unordered, sparse 1D geometric keypoints into ordered, dense 2D semantic maps. Then, to integrate the advantages of both representations, an Adaptive Mixture of Experts (AMoE) module is designed to fuse geometric and semantic representations. It dynamically shifts focus toward more reliable features during the fusion process, allowing the model to handle complex scenes, especially when either modality might be compromised. The fused representation can be adopted into common deep stitching pipelines, delivering significant performance gains over any single feature. Experiments show that UniStitch outperforms existing state-of-the-art methods with a large margin, paving the way for a unified paradigm between traditional and learning-based image stitching. Our project page is available at `https://mmelodyy.github.io/projects/unistitch/`.

Keywords: image stitching, geometric feature, semantic feature

1 Introduction

Image stitching is a fundamental technique in computer vision, enabling the creation of seamless panoramic images from multiple overlapping views. It plays a critical role in a wide range of applications, including virtual reality, autonomous driving, medical imaging, and computational photography.

Existing solutions are generally divided into two categories. Traditional approaches [2,48] (Fig. 1a) leverage hand-crafted geometric features (*e.g.*, SIFT [31]) to establish correspondences and then optimize the desired warps. They perform reliably in structured, texture-rich scenes, yet struggle in low-texture or repetitive environments where feature detection becomes unreliable. In contrast, learning-based methods [35, 36] (Fig. 1b) adopt deep neural networks to extract semantic features, which focus on high-level content understanding of the scene. While this enhances robustness in challenging visual conditions, it also means their learned representations are oriented towards scene understanding rather than capturing geometric structures. Consequently, even in well-structured scenes, where traditional methods perform well, current deep stitching techniques do not consistently demonstrate superiority. This gap naturally raises a critical question: *Where does the future of image stitching lie—in geometric or semantic features?*

In this paper, we answer this by introducing **UniStitch**, a unified image stitching framework from both semantic and geometric features. As shown in Fig. 1c, UniStitch consists of three stages: multi-modality feature alignment, multi-modality feature fusion, and global-to-local warp. Here, we take the most commonly used geometric feature, *i.e.*, keypoint, as the representative and demonstrate how to combine the two totally different modality features.

The first stage is composed of a semantic branch and a geometric branch, which extract multimodal features from images and keypoints, respectively. To

align the discrete keypoints with continuous semantic feature maps, we propose a Neural Point Transformer (NPT) module to encode 1D points into 2D geometric maps. It first transforms shallow keypoints into high-dimensional point features, and then projects them into a structured latent space (*i.e.*, a mesh-like representation) by explicitly reorganizing their spatial relationships. Then, **the second stage** is designed to fuse geometric and semantic features to construct a robust joint representation. To this end, we design an Adaptive Mixture of Experts (AMoE) module that adaptively captures the heterogeneity of multi-modal features and integrates their complementary strengths. Simultaneously, we implement a Latent-space Modality Robustifier (MR) strategy to equip the model with cross-scene robustness, particularly when a certain modality becomes unreliable. **The third stage** predicts the parametric warp following a global-to-local paradigm. Departing from standard practices [37], we introduce a Free-Form Deformation (FFD) module designed to enhance the efficiency of Thin-Plate Spline (TPS) transformations for high-resolution images. This approach significantly reduces VRAM overhead and accelerates inference while preserving precise spatial alignment.

Experiments demonstrate UniStitch achieves SoTA performance across diverse datasets by integrating the strengths of different modality features. Furthermore, it can be combined with different geometric features and make sense, contributing to a pioneering and meaningful step for unified image stitching.

2 Related Work

Here, we review existing image stitching solutions from the perspectives of geometric and semantic features, respectively.

2.1 Traditional Stitching using Geometric Features

Traditional image stitching pipelines typically begin by detecting and matching local geometric features—most classically, keypoints such as SIFT [31]. The seminal Autostitch work [2] demonstrated that a global homography could robustly align images based on such point correspondences. To handle more complex parallax, researchers subsequently extended the warp model beyond global homography to various elastic representations, including mesh-based warps [48,50], thin-plate splines (TPS) [22], superpixel-based deformations [19], and local triangular facets [21]. Alongside this, the feature basis itself has expanded from keypoints to include line segments [23], edges [8], and other structural cues. This evolution not only enhances matching robustness in structured scenes but also actively contributes to preserving geometric shape consistency. These include smoothly transitioning warps from projective to similarity transformation [4,29], enforcing line consistency across the warping process [12,23,27], and imposing a global similarity prior [5]. More recently, some researchers have begun to incorporate higher-level semantic information, notably semantic segmentation, to guide the stitching process. For instance, OBJ-GSP [3] adopts segmentation maps to

extract the contours of any objects and then preserves their structures, while MHW [28] and PRC [20] reorganize geometric correspondences based on segmentation consistency to improve alignment in parallax scenes. This, however, remains a preliminary fusion of geometry and semantics. The semantic prior is used merely as a static guide within the optimization, rather than being jointly encoded and interactively learned with geometric features at the representational level, leaving the two modalities fundamentally isolated.

2.2 Learning-based Stitching using Semantic Features

Driven by the success of deep learning, recent stitching methods have shifted toward end-to-end trainable pipelines. These approaches typically feed image pairs or sequences directly into neural networks, which are trained to predict parametric warps, such as homography [14,33,38], multiple homography [34,45], TPS [17,24,26,36,37,49], or dense optical flow [11,15,18]. By establishing dense, pixel-wise correspondence objectives, these networks learn to extract and align semantic features, gaining notable robustness in challenging scenarios where traditional geometric features are scarce, such as adverse conditions, low-texture, low-light, or repetitive environments [13, 14, 25, 51]. However, this very strength also reveals a limitation: the current research is heavily skewed toward addressing these "hard case", often at the expense of performance in well-structured, geometric-rich scenes where traditional solutions demonstrate reliable performance. The learned representations, focused on high-level content understanding, exhibit a fundamental divergence from explicit geometric structure. Recently, RopStitch [39] further digs into semantic cues by introducing extra content understanding prior, yet still operates predominantly within the semantic feature space. They fall short of explicitly integrating the well-established geometric features inherent to the scene, underscoring a persisting gap: the absence of a unified framework that merges both semantic and geometric features.

3 UniStitch

UniStitch follows a hierarchical three-stage pipeline consisting of multimodal feature alignment, multimodal feature fusion, and global-to-local warp.

3.1 Overview

As illustrated in Fig. 2, we first extract geometric keypoints and descriptors (P_{ref}, P_{tgt}) from a pair of reference and target images (I_{ref}, I_{tgt}). The multimodal feature alignment stage performs intra-modal feature extraction and inter-modal feature alignment. It is composed of two branches, in which the semantic branch extracts high-level semantic maps (denoted as $F_{\text{ref,s}}$, $F_{\text{tgt,s}}$) from the image domain, while the geometric branch leverages a neural point transformer module to convert sparse, unordered keypoints into dense, structured geometric maps (denoted as $F_{\text{ref,g}}, F_{\text{tgt,g}}$). In the multimodal feature fusion stage, we design an

adaptive mixture of experts module with a latent-space modality robustifier that intelligently integrates multimodal cues by weighting their reliability, yielding robust fused features (denoted as $F_{\text{ref,f}}$, $F_{\text{tgt,f}}$). Finally, the global-to-local warp stage adopts a similar warping strategy to StabStitch++ [37], where we incorporate free-form deformation to optimize the vanilla TPS transformation. This crucial modification effectively reduces VRAM consumption and accelerates inference speed while maintaining accurate warping capability.

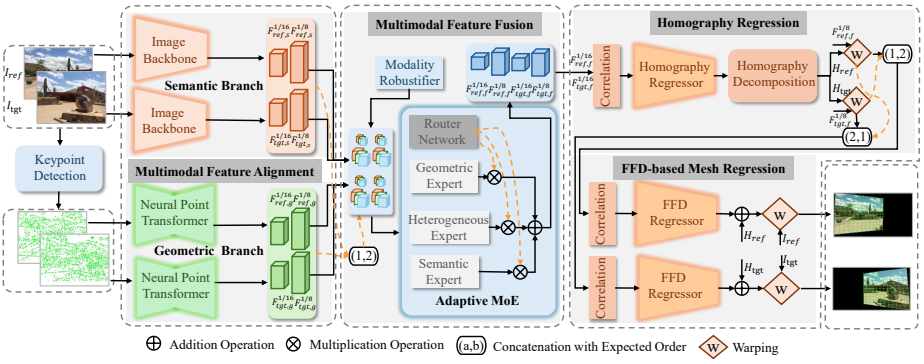

Fig. 2: The overall structure of UniStitch. It takes original images and their keypoints as input, and follows a three-stage pipeline: multimodal feature alignment, multimodal feature fusion, and global-to-local warp.

3.2 Multimodal Feature Alignment

Aligning multimodal features with a unified representation is the first step of UniStitch. To obtain consistent feature representations, we design dual encoding branches based on geometry and semantics to respectively map raw keypoints and image inputs into the common multi-scale feature space.

Semantic Branch Given a pair of images $I_{\text{ref}}, I_{\text{tgt}} \in \mathbb{R}^{3\times H_0\times W_0}$, the semantic branch extracts hierarchical semantic features. We use ResNet-18 [10] as the backbone to construct multi-scale representations at scales $sl \in \{\frac{1}{8}, \frac{1}{16}\}$:

$$F^{sl}_{\text{ref,s}}, F^{sl}_{\text{tgt,s}} = \Phi_{\text{ResNet}}(I_{\text{ref}}, I_{\text{tgt}}). \tag{1}$$

Geometric Branch As shown in Fig. 3, to handle sparse keypoints, we design a neural point transformer module that adopts "Transformation-then-Projection" strategy to generate multi-scale geometric representations. It first encodes keypoint sets $P_{\text{ref}}, P_{\text{tgt}} \in \mathbb{R}^{(2+d)\times n}$ (containing spatial coordinates (x_i, y_i) and descriptors of n points) into latent point feature representations via PointNeXt [42]:

$$P_{\text{ref}}^{\text{feat}}, P_{\text{tgt}}^{\text{feat}} = \Phi_{\text{PointNeXt}}(P_{\text{ref}}, P_{\text{tgt}}), \tag{2}$$

where $P_{\text{ref}}^{\text{feat}}, P_{\text{tgt}}^{\text{feat}} \in \mathbb{R}^{c \times n'}$ represent the c-dimensional latent features of n' downsampled points. To align with semantic feature maps, we assume the geometric feature maps $F_{\text{ref,g}}^{sl}, F_{\text{tgt,g}}^{sl}$ have the same spatial dimensions as $F_{\text{ref,s}}^{sl}, F_{\text{tgt,s}}^{sl}$ (*i.e.*, $H_0 \cdot sl \times W_0 \cdot sl$). In this hypothesis, every element in the geometric maps covers a grid cell of $\frac{1}{sl} \times \frac{1}{sl}$ pixels in the original images. Therefore, we then project $P_{\text{ref}}^{\text{feat}}, P_{\text{tgt}}^{\text{feat}}$ onto zero-initialized grids $\mathbf{V} \in \mathbb{R}^{c \times H_0 \cdot sl \times W_0 \cdot sl}$, in which the keypoint coordinates can be requantized as $\tilde{x}_i = \lfloor x_i \cdot sl \rfloor$ and $\tilde{y}_i = \lfloor y_i \cdot sl \rfloor$. $\lfloor \rfloor$ denotes the floor function. For grid cells containing multiple keypoints, we apply a max-pooling operation to retain the most salient features as:

$$\mathbf{V}[:, \tilde{y}, \tilde{x}] = \max_{k \in \mathcal{C}(\tilde{x}, \tilde{y})} \mathbf{f}_k, \tag{3}$$

where $\mathcal{C}(\tilde{x}, \tilde{y})$ is the keypoint index set within a specific cell and $\mathbf{f}_k$ is the corresponding feature vector in $P_{\text{ref}}^{\text{feat}}, P_{\text{tgt}}^{\text{feat}}$. According to different scales, the re-projected grids following Eq. 3 constitute the desired geometric feature maps $F_{\text{ref,g}}^{sl}$,$F_{\text{tgt,g}}^{sl}$.

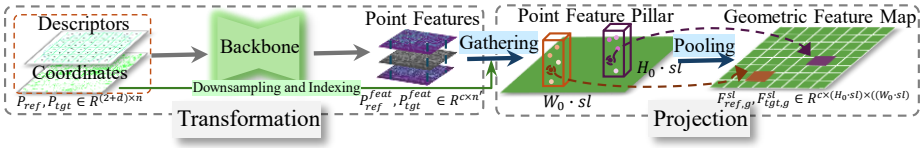

Fig. 3: The details of the neural point transformer module.

3.3 Multimodal Feature Fusion

After feature alignment, we aim to integrate the strengths of different representations: semantic features provide comprehensive global context but are prone to error propagation, while point features offer superior noise resistance despite their non-uniform distribution. The core challenge in achieving a balanced representation lies in harmonizing these modalities without introducing undesirable cross-modal conflicts. We contend that the solution rests upon maintaining a dynamic balance between modal independence and complementarity. To this end, we propose an adaptive mixture of experts module with a latent-space modality robustifier strategy to ensure robust feature aggregation.

Adaptive Mixture of Experts The mixture of experts [47] paradigm is typically applied in multi-task learning to balance the relative significance of competitive objectives. In this work, we adapt this framework for multimodal fusion to

adaptively weigh the contributions of heterogeneous modal features. Our AMoE consists of three residual-based experts—semantic (E_{s}), geometric (E_{g}), and heterogeneous (E_{h})—governed by a linear gated router R. Given a pair of feature maps $(F_{\mathrm{s}}^{sl}, F_{\mathrm{g}}^{sl})$ at scale sl, the router generates a weight vector $\mathbf{w}^{sl} \in \mathbb{R}^3$:

$$\mathbf{w}^{sl} = [w_{\mathrm{s}}^{sl}, w_{\mathrm{g}}^{sl}, w_{\mathrm{h}}^{sl}] = \mathrm{Softmax}\left(R^{sl}\left(F_{\mathrm{s}}^{sl}, F_{\mathrm{g}}^{sl}, F_{\mathrm{s}}^{sl} \oplus F_{\mathrm{g}}^{sl}\right)\right), \tag{4}$$

where $w_{\mathrm{s}}^{sl} + w_{\mathrm{g}}^{sl} + w_{\mathrm{h}}^{sl} = 1$, and $\oplus$ denotes the concatenation operation. Simultaneously, the experts generate corresponding feature maps as:

$$\mathbf{H}_{\mathrm{s}}^{sl} = E_{\mathrm{s}}^{sl}(F_{\mathrm{s}}^{sl}), \quad \mathbf{H}_{\mathrm{g}}^{sl} = E_{\mathrm{g}}^{sl}(F_{\mathrm{g}}^{sl}), \quad \mathbf{H}_{\mathrm{h}}^{sl} = E_{\mathrm{h}}^{sl}(F_{\mathrm{s}}^{sl} \oplus F_{\mathrm{g}}^{sl}). \tag{5}$$

The final fused feature maps $\mathbf{F}_{\mathrm{f}}^{sl}$ is the sum of weighted aggregation as follows:

$$\mathbf{F}_{\mathrm{f}}^{sl} = w_{\mathrm{s}}^{sl}\mathbf{H}_{\mathrm{s}}^{sl} + w_{\mathrm{g}}^{sl}\mathbf{H}_{\mathrm{g}}^{sl} + w_{\mathrm{h}}^{sl}\mathbf{H}_{\mathrm{h}}^{sl}. \tag{6}$$

Modality Robustifier Although the proposed AMoE effectively balances multimodal importance, training on complete, paired multimodal data often leads to over-reliance on modal coupling, neglecting unimodal independence. To prevent performance degradation when one modality becomes unreliable, we propose a latent-space modality robustifier strategy.

Rather than applying augmentation at the input level, which risks disrupting the delicate spatial correspondence between pixels and keypoints, we perform regularization in the latent feature space. It could enhance the robustness in challenging scenes while preserving the capability of multimodal alignment.

To this end, we introduce an additional training phase, *i.e.*, freezing the alignment branches while applying random modal dropout or stochastic Gaussian noise across multi-scale features for all expert branches $r \in \{s, g, h\}$. Denoting $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$, the perturbation process is defined as:

$$\mathbf{H}_r^{sl} = \begin{cases} \mathbf{H}_r^{sl} & \text{if } 1 - p_{\mathrm{drop}} \\ \mathbf{0} & \text{if } p_{\mathrm{drop}} \end{cases} \quad \text{or} \quad \mathbf{H}_r^{sl} = \begin{cases} \mathbf{H}_r^{sl} & \text{if } 1 - p_{\mathrm{noise}} \\ \mathbf{H}_r^{sl} + \boldsymbol{\epsilon} & \text{if } p_{\mathrm{noise}} \end{cases}. \tag{7}$$

Empirically, p_{drop} and p_{noise} are set to 0.25, which forces the model to construct robust representations, especially under partial modal failure or noise.

3.4 Global-to-local Warp

With the fused feature representations, the objective shifts to regressing precise parametric warps for content alignment and shape preservation. Similar to StabStitch++, we adopt a hierarchical global-to-local paradigm to sequentially predict homography and TPS transformations.

Homography Regression In this part, we apply the homography decomposition algorithm [37] to the estimated homography, which is decoupled into $\mathcal{H}_{ref}$ and $\mathcal{H}_{tgt}$. They project the input images onto a virtual middle plane, providing the initial control point offsets for subsequent TPS transformation.

FFD-based TPS Regression With global homography, we predict residual control point offsets for local TPS warps. However, there is a memory bottleneck for TPS transformation in high-resolution. Denoting the projection matrix as $\mathbf{T}$ and output coordinate meshgrid as $\mathbf{M}$, the transformed flows can be approximately expressed as $\mathbf{X}_{\text{flow}} \sim \mathbf{T} \times \mathbf{M}$, in which a massive intermediate caching matrix could be generated.

To avoid it, we reformulate the projection as a compress-then-restore process. We first compress the high-resolution meshgrid $\mathbf{M}$ into a low-resolution $\mathbf{M}^{'}$, with its dimensions empirically set to twice the TPS control point resolution, guaranteeing $\dim(\mathbf{M}^{'}) \ll \dim(\mathbf{M})$. We then obtain the low-resolution transformed flows ($\mathbf{X}^{'}_{\text{flow}} \sim \mathbf{T} \times \mathbf{M}^{'}$), and employ FFD to interpolate the sparse flows back to the original output scale. FFD uses the local support property of B-splines, which ensures that each pixel's deformation can be restored from only a 4×4 local lattice neighborhood as:

$$\hat{\mathbf{X}}_{\text{flow}}(u, v) = \sum_{\alpha=0}^{3} \sum_{\beta=0}^{3} N_{\alpha,3}(u) N_{\beta,3}(v) \mathbf{X}^{'}_{\text{flow}}(u' + \alpha - 1, v' + \beta - 1). \tag{8}$$

(u', v') represents the coordinate of an arbitrary point in $\mathbf{M}^{'}$, while (u, v) denotes the pixel coordinate of any point within a specific cell of $\mathbf{M}$ mapped from the location (u', v') in $\mathbf{M}^{'}$. $N_{.,3}$ denotes the cubic B-spline basis functions, which ensure the interpolated flow is continuous and smooth, and more importantly, $\hat{\mathbf{X}}_{\text{flow}}$ approximates $\mathbf{X}_{\text{flow}}$. Refer to the appendix for more details.

3.5 Objective Function

The objective function $\mathcal{L}$ of UniStitch comprises three terms: content alignment, shape preservation, and expert regularization within AMoE, which is defined as:

$$\mathcal{L} = \mathcal{L}_{align} + w_s \mathcal{L}_{shape} + w_r \mathcal{L}_{reg.}. \tag{9}$$

$\mathcal{L}_{align}$ and $\mathcal{L}_{shape}$ denote the content alignment and shape preservation losses, which follow the same formulation as StabStitch++ for a fair comparison.

$\mathcal{L}_{reg.}$ is the expert regularization loss to ensure the most significant modal features are selected by the AMoE. We define it as:

$$\mathcal{L}_{reg.} = \sum_{r \in \{g,s,h\}} (w_r^{sl} - \bar{w}^{sl})^2 + \lambda_e \sum_{r \in \{g,s,h\}} w_r^{sl} \log(w_r^{sl}), \tag{10}$$

where $\bar{w}^{sl}$ denotes the mean weight across all experts at scale sl. The first term imposes a variance-based penalty to ensure balanced expert utilization, while the second term, the negative entropy, promotes specialization by encouraging experts to become highly selective for specific input modalities. In our implementation, we only constrain it at the $\frac{1}{16}$ scale, *i.e.*, $sl = \frac{1}{16}$.

For bravery, we put the details of $\mathcal{L}_{align}$ and $\mathcal{L}_{shape}$ in the Appendix. The hyperparameters w_s, w_b, and λ_e are empirically set to 10, 0.01, and 0.1.

4 Experiment

4.1 Detail and Dataset

Detail We implemented UniStitch using PyTorch with a single NVIDIA RTX 5090 GPU. The training process followed a two-stage strategy. In the first stage, we trained the model from scratch for 100 epochs without the modality robustifier. In the second stage, we continue training the model for another 100 epochs with the multimodal feature alignment frozen, while introducing the modality robustifier to enhance robustness against modality degradation. The batch size is set to 8, and the initial learning rate is set to 1×10^{-4} with an exponential decay scheduler.

Dataset and Metric Similar to existing learning-based stitching solutions, we train UniStitch on the training set of UDIS-D [35]. To evaluate both in-domain performance and out-of-distribution (OOD) generalization, we test it on the testing set of UDIS-D and classic datasets collected in RopStitch [39], where the former includes 1,106 examples while the latter contains 147 cross-domain samples. As for objective metrics, we adopt mPSNR/mSSIM (as used in existing solutions [15,39]) to measure the average alignment errors in overlapping regions.

4.2 Comparative Experiment

To evaluate the effectiveness of integrating geometric and semantic features, we compare UniStitch against two categories of state-of-the-art methods:

- **Traditional methods using geometric features:** Three representative algorithms (APAP [48], SPW [27], and LPC [12]) are selected, which leverage sparse geometric features to estimate warps.
- **Learning-based methods using semantic features:** Five learning-based stitching algorithms are selected including UDIS [35], UDIS++ [36], DunHuangStitch [32], StabStitch++ [37], and RopStitch [39].

Among the learning-based categories, UDIS and DunHuangStitch only predict the global homography, whereas the others use homography-to-TPS warps. Notably, StabStitch++ is originally a video stitching framework; here, we adapt its spatial warp for comparison with the same number of control points as ours.

For a fair visual comparison, we adopt the average fusion for all methods, highlighting ghosting or blurring artifacts caused by alignment errors.

Quantitative Comparison Tables 1 and 2 provide a comprehensive evaluation of in-domain performance and OOD generalization across the UDIS and classical datasets, respectively. On the UDIS-D dataset, learning-based methods generally surpass traditional geometric algorithms, with the exception of UDIS and DunHuangStitch, due to the limitation of only homography. Conversely, on classic datasets, learning-based models struggle significantly in out-of-domain scenarios,

with most underperforming compared to traditional methods. This suggests that semantic features lack the structural robustness inherent in geometric keypoints when transitioning to unseen environments. Even RopStitch, the latest SoTA using extra frozen semantic priors (from other large-scale datasets [6]), fails to bridge this gap. Unlike existing approaches, UniStitch explicitly combines the cross-scene structural awareness of geometric features with the contextual robustness of semantic features, outperforming existing approaches substantially in both in-domain and OOD settings.

Table 1: Quantitative comparison on the UDIS-D dataset.

	Method	mPSNR↑				mSSIM↑			
		Easy	Moderate	Hard	Average	Easy	Moderate	Hard	Average
1	APAP [48]	26.77	22.88	18.75	22.39	0.868	0.770	0.587	0.726
2	SPW [27]	25.82	21.49	15.85	20.52	0.844	0.693	0.434	0.634
3	LPC [12]	25.01	21.27	17.34	20.82	0.815	0.673	0.485	0.640
4	UDIS [35]	23.53	19.73	17.42	19.94	0.761	0.545	0.376	0.542
5	UDIS++ [36]	27.58	23.75	20.04	23.41	0.880	0.792	0.632	0.755
6	DunHuangStitch [32]	27.19	23.05	19.10	22.61	0.875	0.767	0.564	0.718
7	StabStitch++ [37]	29.92	24.93	20.46	24.63	0.927	0.845	0.664	0.797
8	RopStitch [39]	29.93	24.96	20.60	24.70	0.926	0.845	0.672	0.800
9	Ours	**30.34**	**25.37**	**20.90**	**25.07**	**0.932**	**0.857**	**0.691**	**0.813**

Table 2: Quantitative comparison on classical datasets. $*$ denotes post processing with the iterative adaptation [36].

	Method	mPSNR↑				mSSIM↑			
		Easy	Moderate	Hard	Average	Easy	Moderate	Hard	Average
1	APAP [48]	24.33	19.48	14.47	18.92	0.818	0.687	0.442	0.628
2	SPW [27]	23.83	19.16	14.66	18.75	0.793	0.647	0.445	0.610
3	LPC [12]	23.19	18.67	13.90	18.11	0.761	0.600	0.412	0.573
4	UDIS [35]	20.58	16.27	13.37	16.40	0.660	0.471	0.271	0.454
5	UDIS++ [36]	21.93	17.53	13.83	17.36	0.709	0.526	0.325	0.500
6	DunHuangStitch [32]	22.02	17.44	13.65	17.29	0.712	0.534	0.319	0.501
7	StabStitch++ [37]	23.01	18.08	13.98	17.91	0.751	0.577	0.339	0.534
8	RopStitch [39]	23.40	18.54	14.67	18.44	0.772	0.607	0.387	0.568
9	RopStitch* [39]	25.40	19.79	15.48	19.74	0.839	0.691	0.465	0.645
10	Ours	23.92	18.87	14.93	18.80	0.800	0.643	0.410	0.596
11	Ours*	**26.70**	**20.81**	**16.11**	**20.68**	**0.874**	**0.737**	**0.522**	**0.692**

Qualitative Comparison Figs. 4 and 5 present the stitching results of various algorithms on the UDIS dataset and classic datasets. It can be observed that whether dealing with the in-domain stitching of the UDIS-D dataset or the OOD

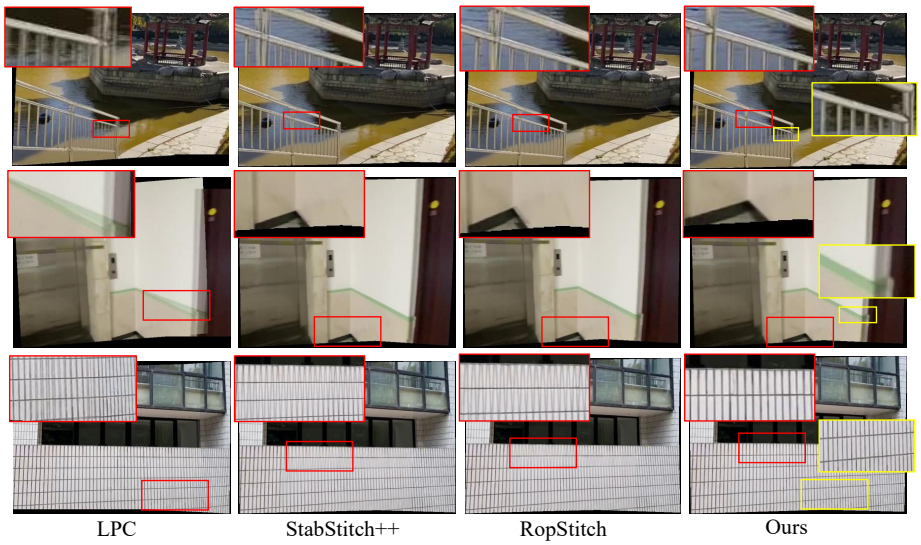

Fig. 4: Qualitative comparison on the UDIS-D dataset.

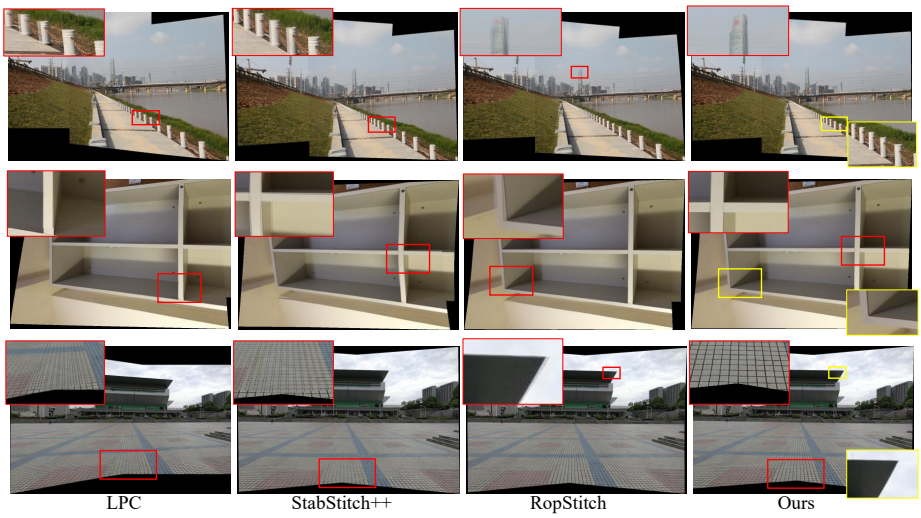

Fig. 5: Qualitative comparison on the classical dataset.

generalization on classic datasets, solely relying on either semantic or geometric representations fails to achieve satisfactory performance.

Benefiting from the advantages of geometric and semantic features, UniStitch achieves superior alignment accuracy and visual coherence in these challenging scenarios. As highlighted by the red boxes, our method effectively eliminates

ghosting and misalignments in complex structures such as floor tiles, railings, and distant buildings, where a single modality feature often struggles.

4.3 Ablation Study

Impact of Network Component As shown in Table 3, we analyze the contribution of each core component. The point-based geometric representation and the image-based semantic representation exhibit distinct advantages: geometric features excel in OOD generalization, while semantic features perform better in in-domain scenarios. By integrating these modalities via the AMoE, we achieve significant mutual complementarity, enhancing performance across both datasets. Furthermore, the MR strategy further sharpens the model's ability to discriminate the contributions from different modalities, highlighting the vast potential of multimodal fusion in image stitching.

Table 3: Ablation study of network component.

	Multimodal Alignment		Multimodal Fusion		UDIS-D	Classic
	Point	Image	AMoE	MR		
1	✓				24.22/0.784	18.02/0.553
2		✓			24.63/0.797	17.91/0.534
3	✓	✓	✓		24.94/0.808	18.56/0.581
4	✓	✓	✓	✓	**25.07**/**0.813**	**18.80**/**0.596**

Impact of Geometric Backbones In Table 4, we evaluate the impact of different point cloud encoders within a point-only baseline configuration. While PointNet shows competitive in-domain results, its OOD performance is inferior to PointNet++ and PointNeXt, underscoring the superior discriminative power of multi-scale feature representations against domain shifts. PointTransformer v3 incorporates sophisticated serialized scanning strategies to capture long-range dependencies, but its serialization of point sets may sacrifice the local positional precision required for fine-grained alignment, resulting in suboptimal convergence and alignment accuracy.

Table 4: Ablation study of different backbones for point feature extraction in the point-only configuration.

	Backbone	UDIS-D	Classic
1	PointNet [40]	**24.47**/**0.793**	17.89/0.542
2	PointNet++ [41]	24.27/0.786	18.00/0.551
3	PointTransformer v3 [46]	23.96/0.778	17.90/0.546
4	PointNeXt [42] (ours)	24.22/0.784	**18.02**/**0.553**

Impact of Point Configuration The impact of point configuration is summarized in Table 5. The model exhibits order-invariance, which is a key factor in the robustness of geometric features, analogous to the scale-invariance found in SIFT-like descriptors. Furthermore, our results emphasize point quality over quantity. In traditional pipelines, excessive outliers degrade alignment, often requiring RANSAC [9] for pruning. Our experiments also follow this phenomenon: matched keypoint pairs yield significantly better results than raw keypoints, demonstrating that reorganizing the keypoints as correspondences is also vital for the learning-based alignment process.

Table 5: Ablation study of different point configurations, including the point order and point quality.

	Point Configuration	UDIS-D	Classic
	Training with Raw Points		
1	– Testing in Fixed Order	24.78/0.803	17.90/0.549
2	– Testing in Random Order	24.76/0.802	17.90/0.549
	Training with Matched Points		
3	– Testing in Fixed Order	**24.91**/**0.808**	**18.48**/**0.577**
4	– Testing in Random Order	**24.91**/**0.808**	**18.48**/0.576

Impact of Fusion Strategies Table 6 investigates various multimodal feature fusion methods. Simple operations like Addition or Concatenation struggle to reconcile the disparate nature of geometric and semantic features. While CSPFusion, a conventional convolutional fusion mode in YOLOv11 [16], integrates multimodal features more effectively, its OOD performance remains suboptimal. Unlike CSPFusion's monolithic design, the proposed AMoE architecture enables differentiated processing of modalities by adaptively weighting expert networks based on the reliability of each modality. This capability is further enhanced by the MR strategy, resulting in substantial gains in OOD generalization.

Table 6: Ablation Study of multimodal feature fusion methods. Refer to Table 3 for the results obtained with single-modality features.

	Fusion Strategy	UDIS-D	Classic
1	Add	24.42/0.790	17.22/0.510
2	Concat	24.61/0.795	17.74/0.538
3	CSPFusion	24.90/0.808	18.52/0.576
4	AMoE	24.94/0.808	18.56/0.581
5	AMoE + MR (ours)	**25.07**/**0.813**	**18.80**/**0.596**

Impact of FFD-based TPS Table 7 compares the performance of vanilla TPS against our FFD-based TPS strategy. Results are averaged over 10 rounds, with peak GPU memory recorded. Compared to vanilla TPS, FFD-based TPS

achieves substantial improvements in both inference speed and VRAM efficiency without compromising alignment accuracy (mPSNR/mSSIM). Moreover, the empirical results demonstrate that FFD-based TPS exhibits excellent resolution scalability: by decoupling deformation complexity from pixel resolution via FFD, UniStitch can effectively scale to high-resolution stitching tasks where vanilla TPS fails due to out-of-memory (OOM) errors.

Table 7: Ablation Study of FFD-based TPS warping strategy. Compared with vanilla TPS, our method offers higher efficiency and lower memory usage without sacrificing alignment accuracy.

Resolution	Warping	Time (s)↓	Peak GPU(GB)	mPSNR↑	mSSIM↑
566×800	Vanilla TPS	0.141	4.81	16.85	0.776
	FFD-based TPS	0.132	2.76	16.86	0.776
1329×2000	Vanilla TPS	0.202	13.99	17.14	0.475
	FFD-based TPS	0.177	8.57	17.12	0.474
2448×3264	Vanilla TPS	-	OOM	-	-
	FFD-based TPS	0.386	22.62	22.71	0.627

4.4 Universality on Geometric Features

To demonstrate the universality of UniStitch, we thoroughly evaluate the benefits of incorporating different geometric features, including SIFT, SURF, ORB, and SuperPoint. As revealed in Table 5, matched keypoints bring more significant performance improvements than raw keypoints. This performance disparity stems from the fact that keypoint matching effectively filters out chaotic spatial redundancies while establishing explicit, high-quality cross-view correspondences to maintain intrinsic topological consistency. Consequently, we apply specific keypoint matching strategies precisely tailored to the characteristics of each geometric feature. Concretely, we employ a traditional nearest-neighbor matching algorithm (*i.e.*, KNN) for handcrafted geometric features, while utilizing advanced attention-based learning matchers (*i.e.*, SuperGlue and LightGlue) for deep learning-based features.

Fig. 6: Ablation study of different geometric features. We draw the matched points on the stitched results to illustrate the keypoint quality. Several local regions are zoomed in to demonstrate the benefit of extra geometric features using our method.

As summarized in Table 8, consistent performance gains across both in-domain (UDIS-D) and out-of-distribution (Classic) datasets confirm that UniStitch robustly leverages diverse geometric priors to complement semantic information. Compared with traditional features, learning-based keypoints demonstrate better results because they can establish more reliable correspondences, especially in challenging scenes (Fig. 6 shows a corresponding example). Besides, from the last two experiments in Table 8, we prove that feature descriptors can be crucial for disambiguating point-based representations.

Table 8: The impact of various geometric features. ‡ denotes experiments that use only keypoint coordinates, without their corresponding descriptors.

Feature Configuration	UDIS-D	Classic
Baseline		
1 – Image-only Configuration	24.63/0.797	17.91/0.534
Tranditional		
2 – SIFT [31] + KNN	24.84/0.805	18.17/0.560
3 – SURF [1] + KNN	24.81/0.804	18.14/0.561
4 – ORB [43] + KNN	24.85/0.806	18.22/0.560
Learning-based		
5 – SuperPoint [7]+SuperGlue [44]	24.85/0.806	18.19/0.568
6 – SuperPoint‡ [7]+LightGlue [30]	24.87/0.807	18.43/0.573
7 – SuperPoint [7]+LightGlue [30] (ours)	**24.91**/**0.808**	**18.48**/**0.577**

5 Conclusion

In this paper, we present UniStitch, a pioneering attempt to integrate geometric and semantic features into a unified image stitching model. By designing multimodal feature alignment and fusion methods, UniStitch effectively bridges the gap between the structural awareness of traditional geometric keypoints and the contextual understanding of learning-based semantic maps. Our architecture ensures that sparse geometric constraints and dense semantic priors complement each other, overcoming the limitations of single-modality approaches in texture-less or unseen environments. Extensive experiments demonstrate the superiority of UniStitch in both in-domain and out-of-distribution scenarios, revealing the promising potential of unifying multimodal features in the field of image stitching. Besides, combined with different geometric features, UniStitch achieves universal and consistent performance gains.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 62502057, 62536002, and 62561160098.

References

1. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: ECCV. pp. 404–417. Springer (2006)
2. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. IJCV **74**(1), 59–73 (2007)
3. Cai, W., Yang, W.: Object-level geometric structure preserving for natural image stitching. In: AAAI. vol. 39, pp. 1926–1934 (2025)
4. Chang, C.H., Sato, Y., Chuang, Y.Y.: Shape-preserving half-projective warps for image stitching. In: CVPR. pp. 3254–3261 (2014)
5. Chen, Y.S., Chuang, Y.Y.: Natural image stitching with the global similarity prior. In: ECCV. pp. 186–201 (2016)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
7. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: CVPRW. pp. 224–236 (2018)
8. Du, P., Ning, J., Cui, J., Huang, S., Wang, X., Wang, J.: Geometric structure preserving warp for natural image stitching. In: CVPR. pp. 3688–3696 (2022)
9. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
11. Jia, Q., Feng, X., Liu, Y., Fan, X., Latecki, L.J.: Learning pixel-wise alignment for unsupervised image stitching. In: ACM MM. pp. 1392–1400 (2023)
12. Jia, Q., Li, Z., Fan, X., Zhao, H., Teng, S., Ye, X., Latecki, L.J.: Leveraging line-point consistence to preserve structures for wide parallax image stitching. In: CVPR. pp. 12186–12195 (2021)
13. Jiang, Z., Li, X., Liu, J., Fan, X., Liu, R.: Towards robust image stitching: An adaptive resistance learning against compatible attacks. In: AAAI. vol. 38, pp. 2589–2597 (2024)
14. Jiang, Z., Zhang, Z., Fan, X., Liu, R.: Towards all weather and unobstructed multispectral image stitching: Algorithm and benchmark. In: ACM MM. pp. 3783–3791 (2022)
15. Jin, H., Nie, L., Lin, C., Feng, X., Zhao, Y.: Pixelstitch: Structure-preserving pixel-wise bidirectional warps for unsupervised image stitching. In: ICCV. pp. 28125–28134 (2025)
16. Khanam, R., Hussain, M.: Yolov11: An overview of the key architectural enhancements. arXiv preprint arXiv:2410.17725 (2024)
17. Kim, M., Lee, Y., Han, W.K., Jin, K.H.: Learning residual elastic warps for image stitching under dirichlet boundary condition. In: WACV. pp. 4016–4024 (2024)
18. Kweon, H., Kim, H., Kang, Y., Yoon, Y., Jeong, W., Yoon, K.J.: Pixel-wise warping for deep image stitching. In: AAAI. vol. 37, pp. 1196–1204 (2023)
19. Lee, K.Y., Sim, J.Y.: Warping residual based image stitching for large parallax. In: CVPR. pp. 8198–8206 (2020)
20. Li, A., Guo, J., Guo, Y.: Image stitching based on semantic planar region consensus. IEEE TIP **30**, 5545–5558 (2021)
21. Li, J., Deng, B., Tang, R., Wang, Z., Yan, Y.: Local-adaptive image alignment based on triangular facet approximation. IEEE TIP **29**, 2356–2369 (2019)

22. Li, J., Wang, Z., Lai, S., Zhai, Y., Zhang, M.: Parallax-tolerant image stitching based on robust elastic warping. IEEE TMM **20**(7), 1672–1687 (2017)
23. Li, S., Yuan, L., Sun, J., Quan, L.: Dual-feature warping-based motion model estimation. In: ICCV. pp. 4283–4291 (2015)
24. Liao, K., Nie, L., Lin, C., Zheng, Z., Zhao, Y.: Recrecnet: Rectangling rectified wide-angle images by thin-plate spline model and dof-based curriculum learning. In: ICCV. pp. 10800–10809 (2023)
25. Liao, K., Wu, S., Wu, Z., Jin, L., Wang, C., Wang, Y., Wang, F., Li, W., Loy, C.C.: Thinking with camera: A unified multimodal model for camera-centric understanding and generation. arXiv preprint arXiv:2510.08673 (2025)
26. Liao, K., Yue, Z., Wu, Z., Loy, C.C.: Mowa: Multiple-in-one image warping model. IEEE TPAMI (2025)
27. Liao, T., Li, N.: Single-perspective warps in natural image stitching. IEEE TIP **29**, 724–735 (2019)
28. Liao, T., Wang, C., Li, L., Liu, G., Li, N.: Parallax-tolerant image stitching via segmentation-guided multi-homography warping. Signal Processing **230**, 109860 (2025)
29. Lin, C.C., Pankanti, S.U., Natesan Ramamurthy, K., Aravkin, A.Y.: Adaptive as-natural-as-possible image stitching. In: CVPR. pp. 1155–1163 (2015)
30. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: Lightglue: Local feature matching at light speed. In: CVPR. pp. 17627–17638 (2023)
31. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60**, 91–110 (2004)
32. Mei, Y., Yang, L., Wang, M., Yu, T., Wu, K.: Dunhuangstitch: Unsupervised deep image stitching of dunhuang murals. IEEE TVCG (2024)
33. Nie, L., Lin, C., Liao, K., Liu, M., Zhao, Y.: A view-free image stitching network based on global homography. Journal of Visual Communication and Image Representation **73**, 102950 (2020)
34. Nie, L., Lin, C., Liao, K., Liu, S., Zhao, Y.: Depth-aware multi-grid deep homography estimation with contextual correlation. IEEE TCSVT **32**(7), 4460–4472 (2021)
35. Nie, L., Lin, C., Liao, K., Liu, S., Zhao, Y.: Unsupervised deep image stitching: Reconstructing stitched features to images. IEEE TIP **30**, 6184–6197 (2021)
36. Nie, L., Lin, C., Liao, K., Liu, S., Zhao, Y.: Parallax-tolerant unsupervised deep image stitching. In: ICCV. pp. 7399–7408 (2023)
37. Nie, L., Lin, C., Liao, K., Zhang, Y., Liu, S., Zhao, Y.: Stabstitch++: Unsupervised online video stitching with spatiotemporal bidirectional warps. IEEE TPAMI (2025)
38. Nie, L., Lin, C., Liao, K., Zhao, Y.: Learning edge-preserved image stitching from multi-scale deep homography. Neurocomputing **491**, 533–543 (2022)
39. Nie, L., Mei, Y., Liao, K., Xu, Y., Lin, C., Xiao, B.: Robust image stitching with optimal plane. IEEE TVCG (2026)
40. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: CVPR. pp. 652–660 (2017)
41. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: NeurIPS. vol. 30 (2017)
42. Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., Ghanem, B.: Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In: NeurIPS. vol. 35, pp. 23192–23204 (2022)
43. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: ICCV. pp. 2564–2571 (2011)

44. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: CVPR. pp. 4938–4947 (2020)
45. Song, D.Y., Um, G.M., Lee, H.K., Cho, D.: End-to-end image stitching network via multi-homography estimation. IEEE SPL **28**, 763–767 (2021)
46. Wu, X., Jiang, L., Wang, P.S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H.: Point transformer v3: Simpler faster stronger. In: CVPR. pp. 4840–4851 (2024)
47. Yang, Y., Jiang, P.T., Hou, Q., Zhang, H., Chen, J., Li, B.: Multi-task dense prediction via mixture of low-rank experts. In: CVPR. pp. 27927–27937 (2024)
48. Zaragoza, J., Chin, T.J., Brown, M.S., Suter, D.: As-projective-as-possible image stitching with moving dlt. In: CVPR. pp. 2339–2346 (2013)
49. Zhang, Y., Lai, Y.K., Nie, L., Zhang, F.L., Xu, L.: Recstitchnet: Learning to stitch images with rectangular boundaries. Computational Visual Media **10**(4), 687–703 (2024)
50. Zhang, Y., Lai, Y.K., Zhang, F.L.: Content-preserving image stitching with piecewise rectangular boundary constraints. IEEE TVCG **27**(7), 3198–3212 (2020)
51. Zhang, Z., Ge, J., Jiang, Z., Zhang, M., Liu, J.: Image stitching in adverse condition: A bidirectional-consistency learning framework and benchmark. In: NeurIPS (2025)