MCNet: Rethinking the Core Ingredients for Accurate and Efficient Homography Estimation

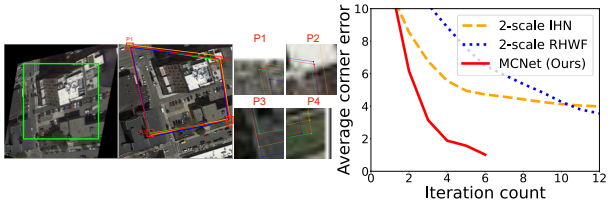
Haokai Zhu[1,2] Si-Yuan Cao[1,2*] Jianxin Hu[2]
Sitong Zuo[4] Beinan Yu[2] Jiacheng Ying[2] Junwei Li[1,2] Hui-Liang Shen[2,3,5]
[1]Ningbo Innovation Center, Zhejiang University [2]College of Information Science and Electronic Engineering, Zhejiang University
[3]Key Laboratory of Collaborative Sensing and Autonomous Unmanned Systems of Zhejiang Province, China
[4]Beijing University of Posts and Telecommunications [5]Jinhua Institute, Zhejiang University
hkzhu.zju@gmail.com cao_siyuan@zju.edu.cn hujianxin@zju.edu.cn,
zuositong1214@bupt.edu.cn {yubeinan,yingjiacheng,lijunwei7788,shenhl}@zju.edu.cn

Abstract

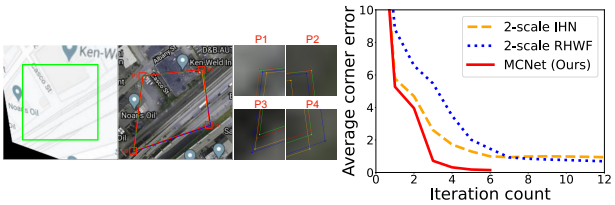
*We propose **M**ultiscale **C**orrelation searching homography estimation **N**etwork, namely MCNet, an iterative deep homography estimation architecture. Different from previous approaches that achieve iterative refinement by correlation searching within a single scale, MCNet combines the multiscale strategy with correlation searching incurring nearly ignored computational overhead. Moreover, MCNet adopts a **F**ine-**G**rained **O**ptimization loss function, named FGO loss, to further boost the network training at the convergent stage, which can improve the estimation accuracy without additional computational overhead. According to our experiments, using the above two simple strategies can produce significant homography estimation accuracy with considerable efficiency. We show that MCNet achieves state-of-the-art performance on a variety of datasets, including common scene MSCOCO, cross-modal scene GoogleEarth and GoogleMap, and dynamic scene SPID. Compared to the previous SOTA method, 2-scale RHWF, our MCNet reduces inference time, FLOPs, parameter cost, and memory cost by 78.9%, 73.5%, 34.1%, and 33.2% respectively, while achieving 20.5% (MSCOCO), 43.4% (GoogleEarth), and 41.1% (GoogleMap) mean average corner error (MACE) reduction. Source code is available at https://github.com/zjuzhk/MCNet.*

1. Introduction

Homography estimation aims to find the projective relationship between two images, which is widely employed in various domains of computer vision and image processing tasks, including image/video stitching [10, 23, 33], multi-modal image fusion [27, 37], video stabilization [13, 26], GPS-denied UAV localization [9, 34, 35], planar object tracking [29, 30, 32], and SLAM [7, 19, 25].



(a) Iterative homography estimation on GoogleEarth.



(b) Iterative homography estimation on GoogleMap.

Figure 1. Visualization of homography estimation with average corner error (ACE) at each iteration of our MCNet, IHN [3], and RHWF [4]. Left 3 images: image pair for homography estimation with the source image $\mathbf{I}_{\mathrm{S}}$ on the left, the target image $\mathbf{I}_{\mathrm{T}}$ on the middle, and the zoomed-in patches centered at each corner on the right. The green polygons denote the ground-truth position of $\mathbf{I}_{\mathrm{S}}$ on $\mathbf{I}_{\mathrm{T}}$. The red, orange, and blue polygons denote the estimated positions using MCNet, IHN, and RHWF, respectively. Right plot: ACEs during 12 iterations. MCNet stops at iteration 6 while 2-scale IHN and 2-scale RHWF at 12.

In the research of deep homography estimation, the first approach [6] employs VGG-style networks to estimate the homography of concatenated image pairs. Subsequent studies [8, 11, 36] make improvements upon this fundamental framework by introducing modified network architectures or cascading multiple similar networks to enhance the accu-

*Corresponding author.

racy. However, the effectiveness of using the cascaded networks for improving the estimation performance is limited [11]. Subsequently, homography estimation approaches [5, 34] based on the IC-LK iterator [1] are introduced. These methods utilize CNN to extract feature maps from images, which are then sent into a pre-computed IC-LK iterator to estimate homography. However, due to the theoretical drawback of the hand-crafted iterator, the estimation performance is limited [3]. To address this issue, IHN [3] adopts an end-to-end iterative homography estimation framework with a trainable iterator that facilitates the updating of correlation, which significantly improves the estimation accuracy. Nevertheless, the effectiveness of the iterative framework within a single scale is hindered as the correlation generated by the down-sampled feature maps own inadequate detailed information. To obtain a better correlation quality, RHWF [4] introduced a homography-guided warping approach, which applies the estimated homography from the previous iteration to the image of the subsequent one before performing feature extraction. By reducing the geometric inconsistencies of the input images, this approach is of better accuracy in the later iterations compared to [3]. However, the iterative feature re-extraction process incurs a substantial computational overhead.

To address the aforementioned issue, we propose a **M**ultiscale **C**orrelation searching **N**etwork, namely MCNet. The main motivation of MCNet is to ensure efficient and effective iteration by combining the iterative search of correlation and multiscale strategy. As the iterative correlation searching process continues, the feature maps of the larger scale are appropriately combined to directly improve the correlation quality. To further boost the estimation performance, we introduce the loss function that produces an increased backward gradient while the loss decreases in the convergence stage, which can dynamically adjust the loss for each sample based on its estimated error. As the L_1 loss of our MCNet can reach a typically low value at the latter training process, we think the homography estimation model has a good chance of reaching a relatively stable capture range. Based on the above observation, we consider raising the backward gradient at this stage by the designed additional loss function, namely **F**ine-**G**rained **O**ptimization (FGO) loss. When incorporating FGO loss with the L_1 loss, MCNet achieves a 32.6% improvement in estimation accuracy on the MSCOCO dataset. Based on the above improvements, our MCNet significantly reduces inference time, FLOPs, parameter cost, and memory cost by 78.9%, 73.5%, 34.1%, and 33.2%, respectively, while achieving 20.5% (MSCOCO [12]), 43.4% (GoogleEarth [34]), and 41.1% (GoogleMap [34]) accuracy improvement, compared to the previous state-of-the-art (SOTA) approach 2-scale RHWF [4]. As illustrated in Fig 1, our MCNet consistently achieves observably more precise estimation as the iteration continues, while the error reduction of 2-scale RHWF [4] and 2-scale IHN [3] generally becomes inconspicuous as the iteration grows. Additionally, our MCNet achieves the average corner error (ACE) below 0.1 for nearly 100% of the test data of the MSCOCO dataset yet yields an inference speed of 30.2fps, indicating its efficiency and accuracy in real-time homography estimation.

On the other hand, the presence of dynamic objects often poses challenges to accurate homography estimation. UDHN [31] uses a mask predictor to depict the plane area in the input images but yields moderate results due to separate predictions on the source and target images. MHN [11] estimates motion flow using PWC-Net [22] to generate masks, but encountered issues with unreliable flow estimation [3]. Subsequently, IHN-mov [3] adopts network architecture to explicitly generate inlier masks, but results in high computational costs. In contrast to the previous methods, the simple introduction of the feature maps of a larger scale in MCNet can implicitly reject outliers by iteratively refining the correlation. This strategy doesn't require any extra computation overhead while achieving superior performance. Experimental results on the SPID dataset [24] show MCNet reduces the mean average corner error (MACE) by 47.1% compared to the previous SOTA 2-scale IHN-mov [3].

In summary, the contributions of our work are as follows:

- We propose MCNet, a multiscale correlation searching network that combines the iterative search of correlation and the multiscale strategy. MCNet achieves SOTA performance on multiple datasets while significantly reducing FLOPs, inference time, parameter cost, and memory cost compared to the previous SOTA approaches.
- Different from the previous L_1 loss, we introduce a **F**ine-**G**rained **O**ptimization (FGO) loss to boost the estimation accuracy. When the model training reaches convergence, cooperating FGO loss with L_1 loss enables the network to have an increased backward gradient as the L_1 loss decreases, enabling a dynamic loss adjustment for different samples, resulting in an accuracy improvement.
- Our network also demonstrates notable performance on challenging datasets with dynamic foreground objects, due to the implicit outlier rejection derived from our multiscale correlation searching framework.

2. Related Work

In this section, we provide a concise review of deep homography estimation and the challenges encountered in the estimation process. For a comprehensive understanding of the basic definitions and principles underlying homography estimation, we refer the readers to relevant literature [38].

Deep homography estimation. DeTone *et al*. [6] first proposed deep homography estimation, employing a VGG-style network to estimate the homography using concatenated image pairs as input. Based on this seminal research,

several studies [8, 11, 36] have subsequently improved the accuracy of homography estimation through cascading or modifying the network. However, the methods mentioned above exhibit subpar performance compared to the Lucas-Kanade iterator-based methods [5, 34] and the subsequent deep iterator-based methods [3, 4], both of which adopt an iterative optimization framework. Chang *et al*. [5] first introduce CLKN employing IC-LK iterator to recurrently estimate homography, and Zhao *et al*. [34] subsequently improves the performance of IC-LK iterator by introducing a loss function to enhance the similarity of feature extracted by CNN. Motivated by the desire to leverage implicit prior knowledge obtained from a vast amount of data, Cao *et al*. [3] proposed an end-to-end trainable iterative estimation framework, which demonstrates substantial accuracy. Building upon their previous work, Cao *et al*. [4] introduced an attention mechanism and homography-guided warping strategy into the recurrent estimation framework, aiming to further enhance the accuracy despite the associated significant computational overhead.

Challenges in homography estimation. The deep iterative homography estimation method based on correlation updates demonstrates good estimation capability [3, 4]. However, experiments reveal minimal error reduction in the later iterations [3, 4], indicating that the computational cost of multiple iterations does not yield significant accuracy improvement. This limitation is attributed to insufficient correlation quality, which hinders the effectiveness of the later iterations. To address this issue, Cao *et al*. [4] proposed a homography-guided warping strategy, which employs estimated homography to warp the image for the next iteration to reduce image deformation before entering into the next iteration. However, this approach incurs a high computational cost due to the need for feature re-extraction at each iteration. On the other hand, real-world scenarios often involve dynamic foreground objects, which violate the assumption of homography estimation and result in poor estimation performance. Some current methods [3, 11, 31] address this by explicitly generating masks to remove outliers. Zhang *et al*. [31] utilizes a mask predictor, but its effectiveness is limited due to separate mask estimation on the source and target images. Le *et al*. [11] employs PWC-Net [22] to estimate optical flow for generating masks. Nonetheless, the estimation of motion flow is prone to instability and potential failure. Cao *et al*. [3] obtains further accuracy improvement by using the correlation as input to generate a mask, but it incurs high computational overhead.

3. Method

The overall schematic diagram of our **M**ultiscale **C**orrelation searching **N**etwork (MCNet) is illustrated in Fig. 2a. MCNet takes the source image $\mathbf{I}_{\mathrm{S}}$ and target image $\mathbf{I}_{\mathrm{T}}$ as input and outputs the estimated homography matrix $\mathbf{M}$. The overall architecture includes multiscale feature extraction and multiscale correlation searching. MCNet conducts one-pass feature extraction for the entire estimation process. Subsequently, the extracted feature maps at different scales are used for multiscale correlation searching. The iteration begins at the lowest-resolution scale and ends at $H \times W$ scale, with each scale running Q iterations.

3.1. Multiscale Feature Extraction

As illustrated in Fig. 2a, the multiscale feature extraction network with shared weights conducts the one-pass extraction of multi-scale feature maps, namely $\mathbf{F}_{\mathrm{S}}^{k_0}$, $\mathbf{F}_{\mathrm{T}}^{k_0}$, $\mathbf{F}_{\mathrm{S}}^{k_0+1}$, $\mathbf{F}_{\mathrm{T}}^{k_0+1}$, $\mathbf{F}_{\mathrm{S}}^{k_0+2}$, and $\mathbf{F}_{\mathrm{T}}^{k_0+2}$, from the source and target images, where k_0 denotes the lowest-resolution scale level. The feature extraction begins with an initial 3×3 convolutional+instance norm+ReLU, followed by a series of basic units (BU), with each BU outputs the feature map of the corresponding scale. As illustrated in Fig. 2b, BU consists of two residual blocks. For the BU that conducts the downsampling operation, the stride of the first residual block is set to 2, otherwise 1. For each scale of the feature map, a 1×1 convolution layer is applied to raise the channel dimension of feature maps used for correlation computations. We employ three basic units to generate feature maps for the three scales: $H/4 \times W/4$, $H/2 \times W/2$, and $H \times W$, with corresponding channel dimensions of 64, 48, and 32, respectively, which are set to be lower than that of RHWF [4] and IHN [3]. We note that a lower-resolution scale of feature maps can also be adopted to further raise the accuracy. Specifically, MCNet only performs feature extraction once throughout the entire estimation process, while the 2-scale IHN [3] involves twice in the scale switching process. Moreover, 2-scale RHWF [4] adopts feature re-extraction at each iteration within each scale, leading to relatively high computation costs. Despite performing only one-pass feature extraction, MCNet outperforms both 2-scale RHWF and 2-scale IHN in terms of estimation accuracy.

3.2. Multiscale Correlation Searching

The core architecture of MCNet is the multiscale correlation searching module, which combines the correlation searching and multiscale strategy. The estimated homography information in the previous iterations, which is carried by the translation $\mathbf{T}_{q_0}$ of the four corner points of the image, is sent into the correlation searching (CS) module first. CS then conducts an iterative correlation search for Q times and then delivers the updated translation ($\mathbf{T}_{q_0+Q}$) to the CS of the next scale. As illustrated in Fig. 2c, correlation searching primarily comprises five components: homography computation, coordinate mapping, correlation computation, correlation decoder (CD), and translation updating. In the following, we will take the **first CS from above** in

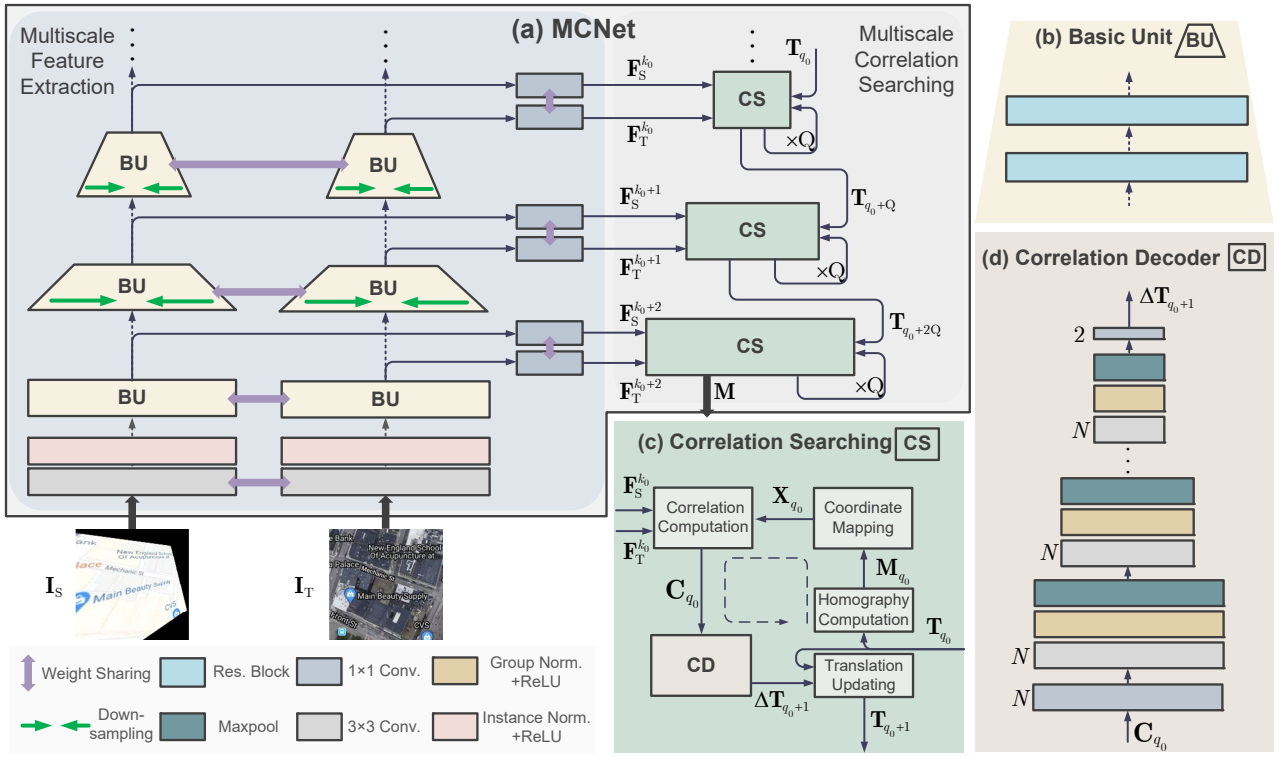


Figure 2. The schematic diagram and detailed architectures of **M**ultiscale **C**orrelation searching **N**etwork, namely MCNet. (a) The overall schematic diagram of MCNet. (b) The architecture of the Basic Unit (BU) module. (c) The architecture of the Correlation Searching (CS) module. (d) The architecture of the Correlation Decoder (CD) module.

Fig. 2a in the q_0 iteration as an example to have a further demonstration of multiscale correlation searching.

Homography Computation. The input accumulated translation $\mathbf{T}_{q_0}$ is first transformed into homography matrix $\mathbf{M}_{q_0}$ using the least square method, which is detailed illustrated in the supplementary material. $\mathbf{M}_{q_0}$ is used for the subsequent pixel-wise coordinate mapping for the correlation computation in this iteration. We note that for the first iteration, the input accumulated translation is set to be 0 horizontally and vertically for the four corner points, representing the homography matrix of the identity transform.

Coordinate Mapping. The coordinate mapping employs the present homography matrix to map the pixel-wise correspondence that associates the source and target feature maps, enabling further computation of correlations. For the coordinate mapping conducted on iteration q_0, we define $\mathbf{X}$ as the coordinate set of source feature map $\mathbf{F}_{\mathrm{S}}$ and $\mathbf{X}'_{q_0}$ as the mapped coordinate set for the target feature map $\mathbf{F}_{\mathrm{T}}$. By defining $\mathbf{x} = (u, v), \mathbf{x} \in \mathbf{X}$ and $\mathbf{x}'_{q_0} = (u'_{q_0}, v'_{q_0}), \mathbf{x}'_{q_0} \in \mathbf{X}'_{q_0}$, the coordinate mapping using $\mathbf{M}_{q_0}$ can be represented as

$$\begin{bmatrix} u'_{q_0} \\ v'_{q_0} \\ 1 \end{bmatrix} \sim \begin{bmatrix} \mathbf{M}_{q_0,11} & \mathbf{M}_{q_0,12} & \mathbf{M}_{q_0,13} \\ \mathbf{M}_{q_0,21} & \mathbf{M}_{q_0,22} & \mathbf{M}_{q_0,23} \\ \mathbf{M}_{q_0,31} & \mathbf{M}_{q_0,32} & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}. \quad (1)$$

Correlation Computation. To achieve higher computational efficiency, we then conduct the on-the-fly correlation computation on feature maps based on the present coordinate mapping. We note that for the two previous correlation-based methods, IHN [3] adopts a direct computation of global correlation volume followed by correlation sampling, while RHWF [4] avoids global correlation computation by directly computing the local correlation. However, RHWF incurs significant computational overhead due to the need for feature re-extraction before each time of correlation computation. By denoting the feature maps of the source image and target image in scale l_0 as $\mathbf{F}_{\mathrm{S}}^{l_0}$ and $\mathbf{F}_{\mathrm{T}}^{l_0}$, the computation of local correlation can be expressed as

$$\mathbf{C}_{q_0}(\mathbf{x}, \mathbf{x}'_{q_0}) = \mathbf{F}_{\mathrm{S}}^{l_0}(\mathbf{x})^{\mathsf{T}} \mathbf{F}_{\mathrm{T}}^{l_0}(\mathcal{A}(\mathbf{x}'_{q_0}, r)), \quad (2)$$

where $\mathbf{x}$ and $\mathbf{x}'_{q_0}$ denote the coordinate position of the source image and the corresponding coordinate of the target image based on the present homography $\mathbf{M}_{q_0}$. $\mathcal{A}(\mathbf{x}'_{q_0}, r)$ denotes the local area with radius r centered at $\mathbf{x}'_{q_0}$, producing the correlation with size $H/4 \times W/4 \times (2r+1) \times (2r+1)$.

Correlation Decoder. We employ a correlation decoder (CD) to decode the input correlation $\mathbf{C}$ into the residual homography prediction, which is parameterized by four-point residual translation $\Delta\mathbf{T}$ as in [3, 4, 6, 11, 21]. As illustrated in Fig. 2d, the CD module consists of multiple elementary blocks, each composed of a 3×3 convolution layer, group normalization, and ReLU activation. The channel dimen-

sion of each elementary block is set to N. For the decoding process in iteration q_0, the correlation $\mathbf{C}_{q_0}$ is first processed by a 1×1 convolution to align the channel dimension and then sent into multiple elementary blocks which finally reduce the spatial dimension of the feature maps to 2×2. A 1×1 convolution layer with an output channel dimension of 2 finally processes the feature maps, producing the predicted residual translation $\Delta\mathbf{T}_{q_0+1}$.

Translation Updating. The estimated residual translation $\Delta\mathbf{T}_{q_0+1}$ is then combined with $\mathbf{T}_{q_0}$ to update the translation $\mathbf{T}_{q_0+1}$ as follows

$$\mathbf{T}_{q_0+1} = \mathbf{T}_{q_0} + \Delta\mathbf{T}_{q_0+1}. \tag{3}$$

We note that we unify the scale of the predicted residual translation of all CDs in MCNet to be the same as the $H \times W$ scale, which makes the correlation searching of multiple feature scales concise.

For the final homography estimation, the translation updated in the last iteration of the $H \times W$ scale is transformed into the ultimate homography prediction $\mathbf{M}$.

3.3. Fine-grained Optimization Loss

Different from previous methods [3, 4, 21] that only employ the L_1 loss for training, we introduce a novel **F**ine-**G**rained **O**ptimization (FGO) loss, which collaborates with L_1 loss to further boost estimation accuracy through introducing additional gradients in the convergence process of the model. As the single L_1 loss of MCNet can reach a typically low value at the latter training process, we infer that the model enters a relatively stable capture range. At this stage, we expect the backward gradient of well-estimated samples to increase as the L_1 loss further decreases. This selectively raises the weights of well-estimated samples based on the training accuracy and dynamically refines its impact during the training process to boost the estimating accuracy. Consequently, we design the FGO loss by employing an inverse proportional function

$$\mathcal{L}_{\mathrm{FGO}}(t) = \begin{cases} 0 & t \geq \alpha \\ -\dfrac{1}{t+\epsilon} & t < \alpha, \end{cases} \tag{4}$$

where ϵ controls the strength of the backward gradient and α controls the timing of incorporating $\mathcal{L}_{\mathrm{FGO}}$. The backward gradient grows as the input t becomes smaller. The overall loss can be represented as

$$\mathcal{L} = \sum_{q=1}^{KQ} \left(\|\mathbf{T}_q - \mathbf{T}_{\mathrm{GT}}\|_1 + \mathcal{L}_{\mathrm{FGO}}(\|\mathbf{T}_q - \mathbf{T}_{\mathrm{GT}}\|_1) \right), \tag{5}$$

where Q denotes the number of iterations at each scale, K the number of overall scales, $\mathbf{T}_q$ the estimated translation at iteration q, and $\mathbf{T}_{\mathrm{GT}}$ the ground truth translation.

4. Experiments

4.1. Implementation Details

We set the number of iterations $Q = 2$ for each scale, the number of scales $K = 3$, the radius $r = 4$ in correlation searching, and the channel dimension $N = 64$ for the correlation decoder. The network is implemented by PyTorch. During the training phase, we use the AdamW optimizer [14] with the maximum learning rate 4×10^{-4}. The number of iterations is set to 120000 with a batch size of 16.

4.2. Datasets and Experiment Settings

We first conduct ablation and evaluation on the MSCOCO dataset [12] as in [3–6, 11, 21, 34]. MSCOCO is a widely-used large-scale image dataset in computer vision tasks, which covers a variety of common scenarios, serving as a fundamental dataset for evaluating homography estimation methods. And then, driven by the practical requirements for navigation and localization, we employ the challenging GoogleEarth and GoogleMap datasets [34] to evaluate the performance under modality inconsistency. Finally, considering that real-world scenes commonly exhibit foreground occlusions that do not accord with the assumptions of homography, we conduct an evaluation on the SPID surveillance dataset [24], which contains dynamic objects and varying illumination that leads to a challenging homography estimation scenario.

Experiment Settings. Similar to the previous works [3–6, 8, 11, 21, 34], the four corner points of input 128×128 images are deformed by random perturbation within the range of $[-32, 32]$ to produce image pair with homography deformation. We use average corner error (ACE) as the metric for homography estimation evaluation, as in [3–6, 8, 11, 21, 34], which computes the MSE of the ground-truth and estimated positions of four corner points.

4.3. Evaluation and Ablation Study on MSCOCO

Ablation Study on MSCOCO. We list the ablation study of MCNet in Table 1. We first evaluate the iteration times for each scale. It is observed that raising the iteration can further produce more accurate homography estimation, with the side effect of a lower inference speed. Thus we chose the $(2, 2, 2)$ combination in the following experiments. We then compare the effectiveness of adopting the scale of a higher resolution under the fixed total iteration time of 6. It is observed that the addition of extra scales in our multiscale correlation searching significantly improves the homography estimation accuracy while requiring negligible time consumption, which indicates the superiority of the multiscale correlation searching framework of MCNet. We then conduct the ablation of the FGO loss in Table 2. We can see that the cooperation of our FGO loss significantly improves the estimation accuracy. The best parame-

Table 1. Ablation study of MCNet. (2, 2, 2) denotes the iteration time for the 3 scales are set to be 2, 2, and 2, respectively.

Experiment	Setting	MACE	Inference Time (ms)
Iteration	**(2, 2, 2)**	**0.031**	**33.1**
	$(4, 4, 4)$	0.023	55.3
	$(8, 8, 8)$	0.022	101.0
Scale	1	0.336	29.6
	2	0.072	30.4
	3	**0.031**	**33.1**

Table 2. Ablation study on the setting of loss.

Loss	ϵ	α	MACE
L1+FGO	0.001	1	0.038
		0.85	0.037
		0.7	0.041
	0.1	1	0.033
		0.85	**0.031**
		0.7	0.033
	0.2	1	0.037
		0.85	0.037
		0.7	0.036
L1	w/o	w/o	0.046

ter combination is $\epsilon = 0.1$ and $\alpha = 0.85$, which is fixed in our following experiments.

To reveal the effect of FGO loss, we train our MCNet using only L_1 loss and evaluate it on the MSCOCO test set. The computed FGO loss gradient, which is normalized for better illustration, is plotted in Fig. 3a, showing that the model assigns larger backward gradients to more accurately estimated samples, thereby improving the estimating performance of the model while ensuring training stability. The zoomed-in result comparison of the original L_1 loss and our FGO loss on MSCOCO are illustrated in Fig. 3b. It is observed that FGO loss achieves a significant improvement on the relatively accurate samples, increasing the fraction of the number of images of ACE<0.01 from 2.5% to 38.3%.

Evaluation on MSCOCO. We evaluate our MCNet on the MSCOCO dataset [12] along with RHWF [4], IHN [3], LocalTrans [21], MHN [11], UDHN [31], DHN [6], CLKN [5], AffNet [18], LFNet [20], PFNet [28], PWC [22], SIFT+ContextDesc+RANSAC [17], SIFT+GeoDesc+RANSAC [16], SIFT+MAGSAC [2], and SIFT+RANSAC [15]. As illustrated in Fig. 4a, we plot the ACE w.r.t the fraction of the number of images for each model following [3–5, 11, 21, 34]. It is observed that our MCNet outperforms all other competitors by a significant margin. Compared to the previous SOTA model, 2-scale RHWF [4], MCNet achieves a 20.5% reduction in MACE and a 31.6% improvement in the fraction of ACE values less than 0.01. Additionally, the proportion of ACE values less than 0.1 estimated by MCNet reaches nearly 100%, demonstrating the superior ability of our model.

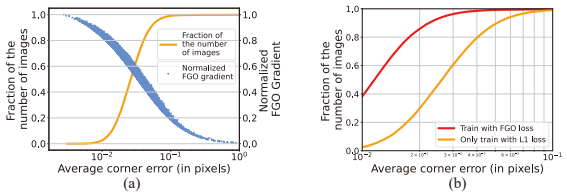


Figure 3. Illustration of FGO loss effectiveness on the MSCOCO test set. (a) Plots of normalized FGO gradients and ACE for each sample, evaluated by MCNet trained only under L_1 loss. (b) ACE plot comparing two MCNet models trained under FGO loss and L_1 loss.

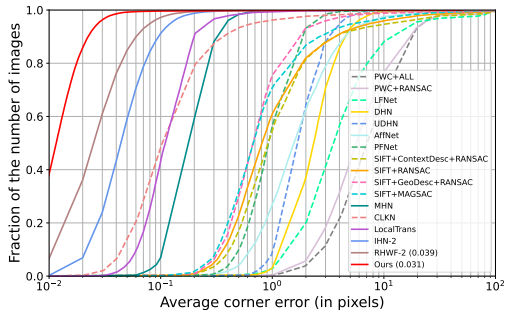
4.4. Evaluation on Cross-Modal Datasets

We further conduct evaluations on GoogleEarth and GoogleMap datasets [34], which contain images with modal inconsistencies. GoogleEarth consists of cross-season satellite images, while GoogleMap includes satellite images and corresponding map images of the same region. We conduct comparison including our MCNet, RHWF [4], IHN [3], MHN+DLKFM [34], MHN [11], DHN+DLKFM [34], DHN [6], CLKN [5], SIFT+MAGSAC [2], SIFT+RANSAC [15], and LK [1]. The corresponding results are plotted in Fig. 4b and Fig. 4c, where we observe that our MCNet outperforms the previous SOTA, 2-scale RHWF, by 43.4% and 41.1% in terms of MACE on GoogleEarth and GoogleMap, respectively.

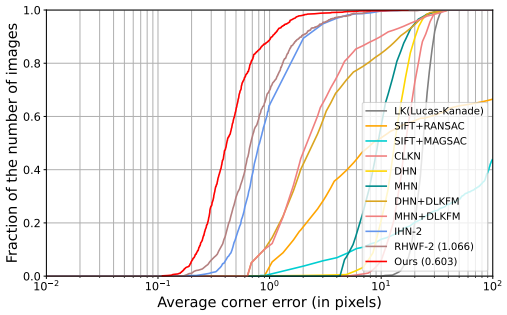
In Fig. 5, we further visualize the correlation at each iteration for the three correlation-based methods IHN[3], RHWF[4], and MCNet on the GoogleMap dataset, by mapping the correlation values to the visual range of $[0, 255]$. In the visualization of correlation, darker regions indicate lower correlation values. It is noticeable that although the correlations of IHN and RHWF continue to update during the iterative estimating process, the improvement is limited primarily due to insufficient feature resolution. The combination of multiscale features and correlation searching in MCNet effectively addresses this issue. As illustrated in the plot, the correlation of MCNet demonstrates considerable improvement as the iteration progresses, resulting in remarkable accuracy enhancement at each iteration and ultimately achieving lower final estimation error within fewer iterations.

4.5. Evaluation on Dataset with Dynamic Objects

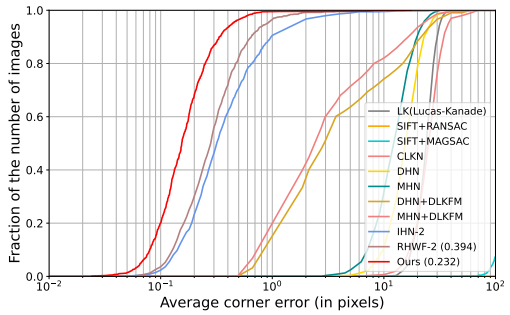
Generalizing homography estimation to real-world scenarios with dynamic foreground objects is a more challenging task. We thus conduct experiments on the SPID surveillance dataset [24] to further evaluate our MCNet. MCNet is compared with IHN [3], UDHN [31], MHN [11], DHN [6], SIFT+MAGSAC [2], and SIFT+RANSAC [15], where UDHN is trained in a supervised manner as in [6] since its unsupervised training on the SPID dataset fails. As illus-



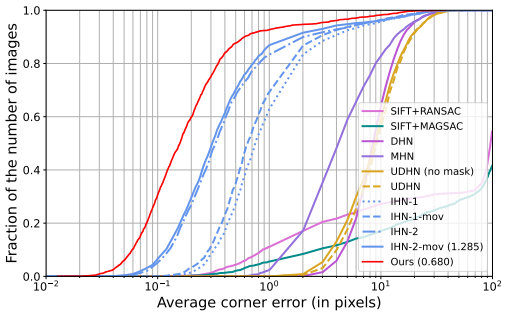
(a) Evaluation on MSCOCO.



(b) Evaluation on GoogleEarth.



(c) Evaluation on GoogleMap.



(d) Evaluation on SPID.

Figure 4. Evaluation of homography estimation methods on MSCOCO, GoogleEarth, GoogleMap, and SPID datasets. The x-axis represents the estimated average corner error (ACE), and the y-axis represents the fraction of data below the corresponding ACE. MSCOCO dataset consists of common RGB images, and GoogleEarth and GoogleMap datasets contain data from cross-modalities. The SPID dataset specifically provides surveillance images that include dynamic foreground objects. The numbers in the brackets next to the legends represent the corresponding mean average corner error (MACE) of the method on the entire dataset.

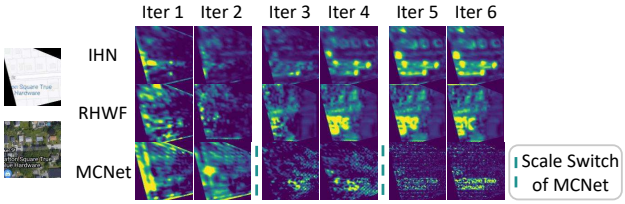


Figure 5. Visualization of correlation produced by our MCNet, IHN[3], and RHWF[4]. The darker regions indicate lower correlation values.

trated in Fig. 4d, MCNet outperforms all other competitors, with a 47.1% reduction of the overall MACE compared to the previous SOTA model 2-scale IHN-mov. Furthermore, we visualize the homography estimation results produced by the aforementioned methods in Fig. 6. It is observed that SIFT+RANSAC and SIFT+MAGSAC produce unsatisfactory homography estimation in some cases. For deep homography estimation methods, DHN, MHN, and UDHN show weak estimation performance influenced by dynamic foreground objects. Even 2-scale IHN-mov that explicitly generates the inlier masks is unable to effectively mitigate the influence of the foreground objects, which proves the effectiveness of our multiscale correlation searching strategy.

It is noticed that our MCNet does not employ explicit mask generation to locate dynamic objects, yet it achieves better estimation performance than 2-scale IHN-mov which explicitly generates inlier masks with higher computational cost. To further explore the performance of our network, we visualized the correlation of MCNet and IHN, namely $\mathbf{C}_{\text{MCNet}}$ and $\mathbf{C}_{\text{IHN}}$, in Fig. 7. The inlier background are visualized in the fused image $\mathbf{I}_{\text{F}}$ by averaging the warped source image $\mathbf{I}_{\text{S}}^{\text{W}}$ and target image $\mathbf{I}_{\text{T}}$. In the visualization of correlation, the dark regions of the correlation visualization of MCNet cover most of the dynamic foreground objects, while the ones of IHN show limited ability to distinguish the foreground object. It indicates that the correlation of our MCNet implicitly generates the attention mechanism of better discrimination under homography constraints. Moreover, the model has acquired the capability to better distinguish between foreground and background under our multiscale correlation searching framework.

4.6. Computational Cost Comparison

We compare MCNet with 2-scale RHWF [4], 2-scale IHN-mov [3], 2-scale IHN [3], and DLKFM [34] algorithms in terms of inference time, FLOPs, parameter cost, and memory cost in Table 3. Additionally, we include the MACEs of these algorithms on the MSCOCO dataset for a comprehensive comparison. The comparison is performed on an NVIDIA Quadro RTX 8000 GPU, with Intel Xeon Silver 4210R CPU @ 2.40GHz, and 64GB of memory.

Among these models, our MCNet not only achieved the lowest MACE, but also demonstrated significant reductions

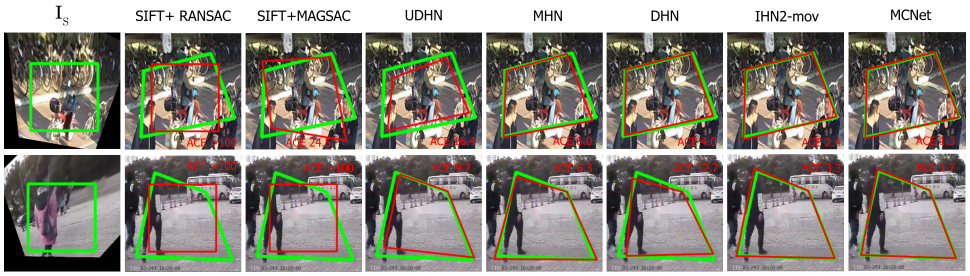


Figure 6. Visualization of the homography estimation results on the SPID dataset with dynamic foreground objects. The green polygon represents the ground-truth location of the source image $\mathbf{I}_S$ on the target image. The red polygon represents the predicted location on the target image estimated by different algorithms. The smaller the relative distance of the polygons and the smaller the ACE the better estimation performance of the corresponding algorithm.

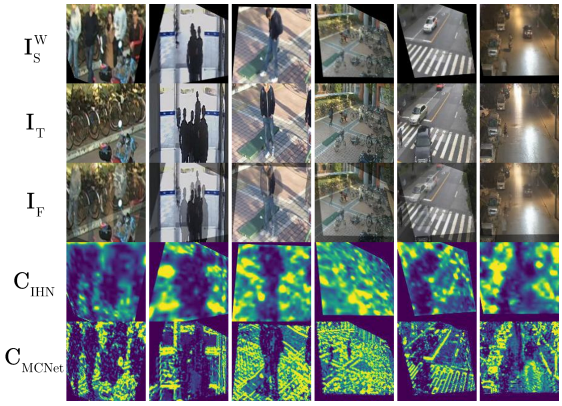


Figure 7. Visualization of correlation produced by our MCNet and IHN [3]. In the 3rd line, the source images are warped and fused with the target one to better illustrate the homography inlier and outlier. Our MCNet produces a much more reasonable correlation that excluding the dynamic objects, whereas the correlation of IHN is ambiguous.

in inference time, FLOPs, parameter cost, and memory cost, which highlights the precision and efficiency of our model. It is observed that compared to the previous SOTA model 2-scale RHWF, our MCNet achieves a 20.5% lower MACE while reducing the inference time, FLOPs, parameter cost, and memory cost by 78.9%, 73.5%, 34.1%, and 33.2%, respectively. Under current settings, MCNet achieves an inference speed of 30.2 fps, indicating its potential in real-time accurate homography estimation.

A Deeper Look into the Efficiency of MCNet. We further investigate the inference time of each module for correlation-based methods, as shown in Table 4. It can be observed that MCNet achieves the lowest inference time in each module. In terms of feature extraction, MCNet significantly reduces the time by 93.1% compared to RHWF2, thanks to its one-pass feature extraction design. Additionally, MCNet saves computation time by eliminating the image warping operation, resulting in zero overhead for image warping. With well-designed efficient multiscale correlation searching, MCNet also achieves notable time reductions of 42.7% and 54.3% in correlation computation and correlation decoding process, respectively.

Table 3. Computational cost comparison.

	Time (ms)	FLOPs (G)	Parameters (M)	Memory (GB)	MACE
DLKFM	380.9	110.51	19.24	4.73	0.550
IHN2	60.1	13.58	1.71	1.89	0.060
IHN2-mov	110.2	36.08	3.40	2.23	0.048
RHWF2	157.1	34.74	1.29	2.53	0.039
MCNet	**33.1**	**9.20**	**0.85**	**1.69**	**0.031**

Table 4. Inference time (ms) of each module.

	Feature extraction	Image warping	Correlation computation	Correlation decoding	Total
IHN2	6.9	1.3	36.5	15.4	60.1
IHN2-mov	6.9	1.3	36.5	65.5	110.2
RHWF2	90.3	16.6	34.2	16.0	157.1
MCNet	**6.2**	**0**	**19.6**	**7.3**	**33.1**

5. Conclusions

We have proposed the homography estimation network based on multiscale correlation searching, namely MCNet. Different from the previous iterative homography estimation methods that raise accuracy by conducting the iteration within a single scale, MCNet achieves accurate and efficient homography estimation by combining the multiscale information with correlation searching. We have also devised a loss function, named FGO loss, which can further facilitate the network training as it approaches convergence, improving estimation accuracy without introducing any additional computation. Experimental results show that adopting the above two strategies into MCNet can achieve high-accuracy homography estimation while preserving the inference efficiency, which realizes SOTA performance together with a much lower computational cost compared to the previous approaches.

Acknowledgments. This work was supported in part by the National Key Research and Development Program of China under grant 2023YFB3209800, in part by the National Natural Science Foundation of China under grant 62301484, and in part by the Natural Science Foundation of Zhejiang Province under grant D24F020006.

References

[1] Simon Baker and Iain Matthews. Lucas-Kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, 2004. 2, 6

[2] Daniel Barath, Jiri Matas, and Jana Noskova. MAGSAC: marginalizing sample consensus. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10197–10205, 2019. 6

[3] Si-Yuan Cao, Jianxin Hu, Zehua Sheng, and Hui-Liang Shen. Iterative deep homography estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1879–1888, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[4] Si-Yuan Cao, Runmin Zhang, Lun Luo, Beinan Yu, Zehua Sheng, Junwei Li, and Hui-Liang Shen. Recurrent homography estimation using homography-guided image warping and focus transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9833–9842, 2023. 1, 2, 3, 4, 5, 6, 7

[5] Che-Han Chang, Chun-Nan Chou, and Edward Y Chang. CLKN: Cascaded lucas-kanade networks for image alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2213–2221, 2017. 2, 3, 6

[6] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016. 1, 2, 4, 5, 6

[7] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *Proceedings of the European Conference on Computer Vision*, pages 834–849. Springer, 2014. 1

[8] Farzan Erlik Nowruzi, Robert Laganiere, and Nathalie Japkowicz. Homography estimation from image pairs with hierarchical convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 913–920, 2017. 1, 3, 5

[9] Hunter Goforth and Simon Lucey. GPS-denied UAV localization using pre-existing satellite imagery. In *2019 International Conference on Robotics and Automation*, pages 2974–2980. IEEE, 2019. 1

[10] Heng Guo, Shuaicheng Liu, Tong He, Shuyuan Zhu, Bing Zeng, and Moncef Gabbouj. Joint video stitching and stabilization from moving cameras. *IEEE Transactions on Image Processing*, 25(11):5491–5503, 2016. 1

[11] Hoang Le, Feng Liu, Shu Zhang, and Aseem Agarwala. Deep homography estimation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7652–7661, 2020. 1, 2, 3, 4, 5, 6

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer, 2014. 2, 5, 6

[13] Shuaicheng Liu, Lu Yuan, Ping Tan, and Jian Sun. Bundled camera paths for video stabilization. *ACM Transactions on Graphics*, 32(4):1–10, 2013. 1

[14] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[15] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 6

[16] Zixin Luo, Tianwei Shen, Lei Zhou, Siyu Zhu, Runze Zhang, Yao Yao, Tian Fang, and Long Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *Proceedings of the European Conference on Computer Vision*, pages 168–183, 2018. 6

[17] Zixin Luo, Tianwei Shen, Lei Zhou, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. ContextDesc: Local descriptor augmentation with cross-modality context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2527–2536, 2019. 6

[18] Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Repeatability is not enough: Learning affine regions via discriminability. In *Proceedings of the European Conference on Computer Vision*, pages 284–300, 2018. 6

[19] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. 1

[20] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. LF-Net: Learning local features from images. *arXiv preprint arXiv:1805.09662*, 2018. 6

[21] Ruizhi Shao, Gaochang Wu, Yuemei Zhou, Ying Fu, Lu Fang, and Yebin Liu. LocalTrans: A multiscale local transformer network for cross-resolution homography estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14890–14899, 2021. 4, 5, 6

[22] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8934–8943, 2018. 2, 3, 6

[23] Richard Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 2(1):1–104, 2006. 1

[24] Dan Wang, Chongyang Zhang, Hao Cheng, Yanfeng Shang, and Lin Mei. SPID: Surveillance pedestrian image dataset and performance evaluation for pedestrian detection. In *Asian Conference on Computer Vision*, pages 463–477. Springer, 2016. 2, 5, 6

[25] Xi Wang, Marc Christie, and Eric Marchand. TT-SLAM: Dense monocular slam for planar environments. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11690–11696. IEEE, 2021. 1

[26] Weiqing Yan, Yiqiu Sun, Wujie Zhou, Zhaowei Liu, and Runmin Cong. Deep video stabilization via robust homography estimation. *IEEE Signal Processing Letters*, 2023. 1

[27] Jiacheng Ying, Hui-Liang Shen, and Si-Yuan Cao. Unaligned hyperspectral image fusion via registration and interpolation modeling. *IEEE Transactions on Geoscience and Remote Sensing*, 2021. 1

[28] Rui Zeng, Simon Denman, Sridha Sridharan, and Clinton Fookes. Rethinking planar homography estimation using

perspective fields. In *Asian Conference on Computer Vision*, pages 571–586. Springer, 2018. 6

[29] Xinrui Zhan, Yueran Liu, Jianke Zhu, and Yang Li. Homography decomposition networks for planar object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3234–3242, 2022. 1

[30] Haoxian Zhang and Yonggen Ling. HVC-Net: Unifying homography, visibility, and confidence learning for planar object tracking. In *Proceedings of the European Conference on Computer Vision*, pages 701–718. Springer, 2022. 1

[31] Jirong Zhang, Chuan Wang, Shuaicheng Liu, Lanpeng Jia, Nianjin Ye, Jue Wang, Ji Zhou, and Jian Sun. Content-aware unsupervised deep homography estimation. In *Proceedings of the European Conference on Computer Vision*, pages 653–669. Springer, 2020. 2, 3, 6

[32] Zhicheng Zhang, Shengzhe Liu, and Jufeng Yang. Multiple planar object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23460–23470, 2023. 1

[33] Qiang Zhao, Yike Ma, Chen Zhu, Chunfeng Yao, Bailan Feng, and Feng Dai. Image stitching via deep homography estimation. *Neurocomputing*, 450:219–229, 2021. 1

[34] Yiming Zhao, Xinming Huang, and Ziming Zhang. Deep Lucas-Kanade homography for multimodal image alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15950–15959, 2021. 1, 2, 3, 5, 6, 7

[35] Hang Zhong, Yaonan Wang, Zhiqiang Miao, Ling Li, Shuangwen Fan, and Hui Zhang. A homography-based visual servo control approach for an underactuated unmanned aerial vehicle in GPS-Denied environments. *IEEE Transactions on Intelligent Vehicles*, 8(2):1119–1129, 2022. 1

[36] Qiang Zhou and Xin Li. STN-homography: Direct estimation of homography parameters for image pairs. *Applied Sciences*, 9(23):5187, 2019. 1, 3

[37] Yuan Zhou, Anand Rangarajan, and Paul D Gader. An integrated approach to registration and fusion of hyperspectral and multispectral images. *IEEE Transactions on Geoscience and Remote Sensing*, 58(5):3020–3033, 2019. 1

[38] Barbara Zitova and Jan Flusser. Image registration methods: A survey. *Image and Vision Computing*, 21(11):977–1000, 2003. 2