TransFill: Reference-guided Image Inpainting by Merging Multiple Color and Spatial Transformations

Yuqian Zhou[1], Connelly Barnes[2], Eli Shechtman[2], Sohrab Amirghodsi[2]
[1]IFP, UIUC, [2]Adobe Research

Figure 1: Results of our reference-guided inpainting for user-provided images. We show multiple practical applications like replacing and removing foreground people and objects. Each triad shows the target image with the hole, the source image used as a reference, and the inpainting result. Our method has strong performance and addresses challenging real-world issues such as parallax, 90 degree image rotations, and lighting inconsistency between the source and target images.

Abstract

Image inpainting is the task of plausibly restoring missing pixels within a hole region that is to be removed from a target image. Most existing technologies exploit patch similarities within the image, or leverage large-scale training data to fill the hole using learned semantic and texture information. However, due to the ill-posed nature of the inpainting task, such methods struggle to complete larger holes containing complicated scenes. In this paper, we propose TransFill, a multi-homography transformed fusion method to fill the hole by referring to another source image that shares scene contents with the target image. We first align the source image to the target image by estimating multiple homographies guided by different depth levels. We then learn to adjust the color and apply a pixel-level warping to each homography-warped source image to make it more consistent with the target. Finally, a pixel-level fusion module is learned to selectively merge the different proposals. Our method achieves state-of-the-art performance on pairs of images across a variety of wide baselines and color differences, and generalizes to user-provided image pairs.

1. Introduction

Image inpainting is an image restoration task where the goal is to fill in specific regions of the image while making the entire image visually realistic. The regions to be filled are called hole regions, and could contain undesired foreground objects or small distracting elements, or corrupted regions of the image. Much research has been devoted to improving image inpainting either by image self-similarity (e.g. [3]) or deep generative models (e.g. [67, 60, 66]). Such methods synthesize realistic semantics and textures by reusing similar patches from non-hole regions or learning from large collections of images, respectively. However, those methods still struggle in cases when holes are large, or the expected contents inside hole regions have complicated semantic layout, texture, or depth.

These problems can be addressed if there happens to be a second reference image of the same scene that exposes some desired image content that can be copied to the hole. This task is referred to as *reference-guided* image inpainting in the literature [37], but this topic is less explored. In our paper, we call the image with the hole indicated for removal the *target image*. In general, there could be multiple other *source images* used as references. These could be taken by the photographer for the same scene after objects have moved or the photographer moved the camera to a different viewpoint to expose the background. Alternatively, a source image could be collected from the Internet [57]. If one such source image contains new desired appearance for the target hole region, then we can copy the pixels from the source to fill in the target hole regions. In this paper we assume that the user has identified a particular source image with the new desired appearance, so we refer to this as *the source* image. We imagine that dedicated apps might be created for aiding the photographer in this process, or for automatically retrieving suitable such source images from the Internet.

Although the reference source image makes the inpainting task easier, reference-guided inpainting is still quite challenging for several reasons. First, the hole regions could be very large, which makes the task of guessing the pixel colors in the hole region less well-posed. Second, we wish for our task to be as general as possible, so we allow an uncalibrated camera to freely translate to different 3D positions for the source and target image, because this can allow the photographer to intentionally reveal regions behind a foreground object to be removed. Such translations, however, can induce large parallax, which cannot be modeled in image space by a simple 2D warp such as a global homography. Unlike video inpainting or multi-view Structure-from-Motion (SfM), we assume the system will not have access to more than two photos. Thus, it is harder in our setting to reliably estimate 3D structures, depth, and point correspondences. Third, depending on the camera and photography setup, the photographs may have substantially different exposure, white balance, or lighting environment, and if one photograph comes from the Internet, then it will have different camera response curves. Existing methods based purely on warping cannot resolve the resulting complex issues of color matching. Finally, there may exist regions in the source image that do not exist after warping due to pixels being out of the image or occluded.

To address these challenges, we propose a multi-homography fusion pipeline combined with deep warping, color harmonization, and single image inpainting. We address the issue of parallax by assuming that there may be multiple depth planes inside the hole. Loosely inspired by recent work on multiplane images [9, 75, 33, 53], we propose multiple homographic registrations of the source image to the target, each corresponding to an assumption that the scene geometry lies on a different 3D plane. Given a target and a source image, we first estimate the matched feature points between the two images, cluster the inliers according to their estimated depths in the target image, and for each cluster estimate a single homography to perform an initial image registration. We call each of these candidate alignment images a *proposal*. For each proposal, we then tackle the challenge of color matching by using a deep bilateral color transformation, and we address parallax issues by refining the warp using a learned per-pixel spatial transformation. We then merge all the transformed source image proposals by learning a set of fusion masks. Finally, we address the last challenge regarding regions which do not exist in the source image by using a state-of-the-art single image inpainting method to complete missing regions, and learn to merge it as well.

In summary, the main contributions of our method are: (1) We propose TransFill, a multi-homography estimation pipeline to obtain multiple transformations of the source image, where each aligns a specific region to the target image; (2) We propose to learn a color and spatial transformer to simultaneously perform a color matching and make a per-pixel spatial transformation to address any residual differences after the initial alignment; (3) We learn weights suitable for combining all final proposals with a single image inpainting result.

2. Related Work

Image inpainting. Inpainting research can be divided into two categories: traditional methods that work by propagating colors or matching patches, and deep methods that learn semantics and texture from large image datasets.

Some traditional methods propagate pixel colors by anisotropic diffusion [4] or solving PDEs [2]. Such methods work well for thin hole regions but as the hole regions grow larger they tend to result in over-blurring. Patch-based image inpainting methods work by finding similar matches elsewhere in the image and copying the resulting texture [56, 3]. Those methods tend to result in high-quality texture but may give implausible structure and semantics.

Our work is more closely related to deep models for inpainting that use a single image. Context encoders analyze the surroundings of the hole [39], local and global discriminators [17] can improve local texture and overall image layout, and partial [29] and gated convolutions [67] can reduce artifacts from filter responses at the hole boundary.

More recently, some deep methods have focused on inferring other information first: these can be roughly categorized into using edges [34], segmentation masks [47], low-frequency structures [43, 27], and other possible maps like depth. The ill-posed nature [74] of single-image inpainting makes it challenging to complete larger holes and higher-resolution images. Recent works demonstrate neural networks can generate high-resolution images [63, 69, 65], but for large holes, these methods can still generate results that appear semantically implausible or have artifacts in the fine-scale texture. Since our method has a source reference image, we can better establish consistency with the ground truth image by learning appropriate spatial and color transformations for a source image patch.

Video inpainting. A few classical works in this area are Wexler *et al*. [56] and Granados *et al*. [13], which globally optimize patch-based energies, and Newson *et al*. [35], which estimates multiple homographies using a piecewise planar assumption for the scene. Xu *et al*. [61] estimates the optical flow to learn the pixel warping field. Recently, the Onion-Peel Network (OPN) [37] leverages non-local designs inside the network, making it feasible to apply multi-source inpainting for a larger temporal range. Lee *et al*. [25] proposed a Copy-and-Paste Network to learn the alignment of consecutive frames for video inpainting. Zhao *et al*. [73] reuse contents from an unrelated image for a reference-based inpainting. Their method is based on only a single affine transform, which we show is not enough in our experiments, and exhibits residual color and geometric incompatibilities that are problematic in our multi-view scenario. Xue *et al*. [62] is a specialized method designed to remove reflective or occluding elements near the camera such as fences.

Image harmonization. Image harmonization refers to matching the color distribution and appearance when compositing a foreground from one image on a background from another image. Traditional methods transfer color statistics locally and globally [41, 42, 50] and use gradient-domain based blending [40, 20, 51]. Digital photomontage [1] also demonstrated copy-and-paste workflows that can change the appearance of a foreground subject. Unlike our method, photomontage required user input and assumes the photographs have been aligned. Recently, CNN-based harmonization models [76, 59] are emerging, including methods involving segmentation masks [52] for region selection, and discriminators for domain verification [6]. Deep bilateral filtering has also been used to better preserve edges and details while transforming image color space [12, 54]. Our work is the first to integrate harmonization with a neural network for reference-guided inpainting. We apply a deep bilateral color transformation to address color inconsistencies while preserving edges.

Image alignment. Image alignment or registration involves placing multiple images in the same coordinate system. It is widely used for video stabilization [30], image stitching [24, 14], and serves as an important pre-processing step for many video and image applications like face analysis. Homography warping is a widely used global parametric method. Sparse local features like SIFT [31] can be matched either using nearest neighbour, or deep models like OANet [70] and SuperGlue [45], and the resulting correspondences can be used to estimate warping models. Recently, deep models have been explored to directly learn homography parameters [7, 71, 36], demonstrating their advantages on low-light and low-texture images.

Issues of parallax due to content at different depths can be better addressed by mesh-based warping [30, 26, 28] or pixel-wise dense optical flow [15, 58, 48, 49, 55, 18]. Liu *et al*. proposed the Content Preserving Warp (CPW) [28] to maintain the rigidity of motions. Recently, Ye *et al*. proposed deep meshflow [64] to make mesh estimation more robust on different scenes. Due to the sparsity of the mesh, image contents can be better retained while warping. However, optical-flow based methods can provide greater flexibility in permitted motions. Our pipeline uses multiple global homographies followed by per-pixel warping fields to combine the advantages of various alignment methods.

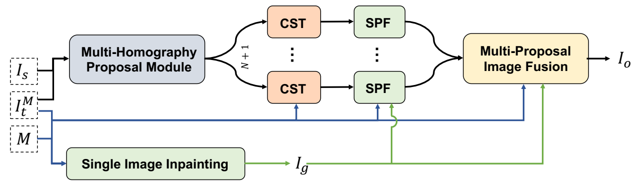

Figure 2: System pipeline. Given the target image I_t^M masked by an associated binary hole image M, and a single source image I_s, we first propose multiple global homographies using the multi-homography proposal module, and locally adjust color and spatial misalignments in each proposal using our Color-Spatial Transformer (CST). Then we merge each proposal with the output I_g from a single-image inpainting model using Single-Proposal Fusion (SPF), and finally selectively blend all the proposals.

3. Method

We will first give an overview of our pipeline. Suppose we are given a target image $I_t \in \mathbb{R}^{W \times H \times 3}$, an associated mask $M \in \mathbb{R}^{W \times H \times 1}$, and a single source image $I_s \in \mathbb{R}^{W_s \times H_s \times 3}$. Note that M indicates the hole regions with value one, and elsewhere with zero. The masked target image is then denoted by $I_t^M = (1 - M) \odot I_t$. We assume there is sufficient overlap in content between the two images especially nearby (but not necessarily within) the masked regions. Our task is to generate contents inside the masked regions of I_t by effectively reusing contents of I_s. More specifically, we wish to geometrically align I_s with I_t in the vicinity of the hole region globally and locally, and adjust any color inconsistency. We fill any regions that are occluded or outside the image using a state-of-the-art single image inpainting method.

Our pipeline follows four steps as shown in Figure 2. It includes an initial registration using multiple homography proposals, per-pixel color and spatial transformations for each proposal, single-proposal fusion and multi-proposal fusion, as introduced in the following sections.

3.1. Multi-homography Proposals

In this stage, we first globally warp the source image I_s to align it with the masked target image I_t^M. Provided the contents inside the hole region occur at multiple depth planes, or the camera motion is not a simple rotation, a single homography is not sufficient to perfectly align the source and target image [10]. Therefore, we propose to es-

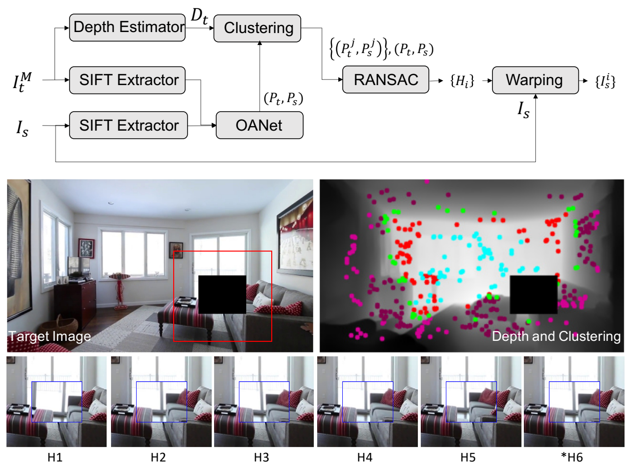

Figure 3: Multi-homography Proposal Module. We compute the monocular depth D_t of the non-hole region I_t^M, and cluster the feature matching points into N sub-groups using the depth values. Each estimated homography H_i will align different regions within the hole. $^*H_6$ indicates a homography estimated using all the points.

timate multiple homography matrices to transform I_s. Ideally, each homography-transformed I_s can align with I_t within a specific image depth level range or local spatial area, as shown in Figure 3.

To obtain different transformation matrices, we first extract SIFT [32] features from I_t^M and I_s, and feed all the extracted feature points and their descriptors into a pre-trained OANet [70] for outlier rejection. The lightweight OANet efficiently establishes the correspondences between I_t^M and I_s by considering the order of the points in the global and local context. OANet outputs the inliers forming a point set P_t in I_t^M, and its corresponding matched point set P_s in I_s. To consider different possible depth planes within and nearby the hole region, we are inspired by the Multi-Plane Image (MPI) [75] idea for scene synthesis. We estimate the depth map D_t from I_t^M using a deep learning based monocular depth estimator [16] , and record the depth value for each point in P_t. We then cluster those points into a partition of N subsets $\{P_t^j\}, j \in [1, N]$ by their depth values using an agglomerative clustering method [21], where $P_t = \cup_{j=1}^N P_t^j$. The corresponding matched points in P_s are used to form the subsets $P_s = \cup_{j=1}^N P_s^j$ accordingly.

For each subset's pairs of points (P_t^j, P_s^j), we estimate a single homography using RANSAC [8]. By further including the homography estimated from the full set of points (P_t, P_s), we obtain $N+1$ homography matrices overall. We denote them by $H_i, i \in [1, N+1]$. Finally, we transform the source image I_s using the estimated H_i, and obtain a set of warped source images $\{I_s^i\}$, where $I_s^i \in \mathbb{R}^{W \times H \times 3}, i \in [1, N+1]$. We set $N = 5$ in our experiments.

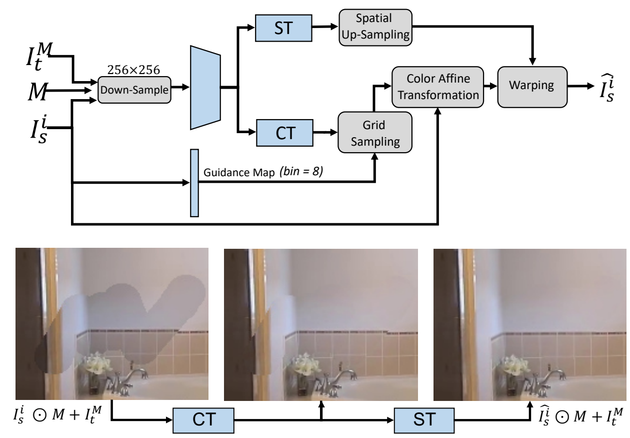

Figure 4: Structure of the Color-Spatial Transformer Module. I_s^i will first go through a Color Transformer (CT), and then a Spatial Transformer (ST) to obtain a refined source image $\hat{I}_s^i$. The bottom row shows examples of the refinement stages. Blocks with blue color indicate there are learned parameters, otherwise they are parameter-free.

3.2. Color-Spatial Transformation (CST) Module

The global homography-warped source image sets $\{I_s^i\}$ are regarded as the initialization of the warping of I_s. However, as shown in Figure 3 and 4, while directly compositing I_s^i and I_t^M using $I_s^i \odot M + I_t^M$, due to the possibly inaccurate homography estimation or challenges of large parallax, there may be small misalignments inside and near the hole region, especially along the hole boundary. Additionally, the composite image may suffer from color and exposure differences. Therefore, we propose another refinement step that we call a Color-Spatial Transformer (CST). This simultaneously adjusts the color and alignment for each I_s^i. The structure of CST is illustrated in Figure 4. I_s^i will first go through a Color Transformer (CT), and then a Spatial Transformer (ST) to obtain a refined source image $\hat{I}_s^i$.

In our design of the color and spatial transformers, we would like to retain the texture details and the rigidity of the source image contents. Additionally, we prefer the color transformation and warping operations to be decoupled and not have to use auxiliary losses for each component. Inspired by deep bilateral filtering [12] and Spatial-Transformer Network (STN) [19], we propose to learn the transformations in a lower resolution, and obtain the full-resolution coefficients using up-sampling. Specifically, given I_s^i, I_t^M and M, we down-sample them to 256×256 to obtain $I_s^i \downarrow$, $I_t^M \downarrow$ and $M \downarrow$. Then we compute the high-level features $u_s^i = B(I_s^i \downarrow, I_t^M \downarrow, M \downarrow)$ using a shared network B. After that, the color and spatial transformation coefficients will be learned by the CT and ST sub-networks.

Color Transformation (CT). To transform the color in RGB space of I_s^i to I_{sc}^i, we learn an affine transformation

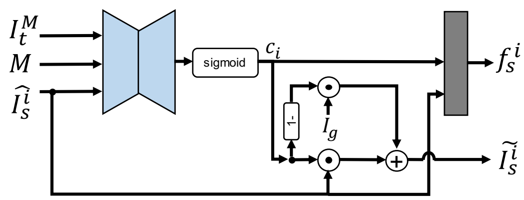

Figure 5: Single-Proposal Fusion (SPF) module. This takes I_t^M, M, a single $\hat{I}_s^i$ and I_g as inputs, where I_g is the result of a single image inpainting method. SPF outputs a confidence map c_i, the merged $\tilde{I}_s^i$, and the packed features f_s^i.

with parameters $A_c^i = [K_c^i \quad b_c^i] \in \mathbb{R}^{W\times H\times 3\times 4}$. Formally, for each pixel at location p, $I_{sc}^i(p) = K_c^i(p) I_s^i(p) + b_c^i(p)$, where $K_c^i(p) \in \mathbb{R}^{3\times 3}$ and $b_c(p) \in \mathbb{R}^{1\times 3}$. To better preserve the edges and textual details, we adopt deep bilateral filtering [12]. Specifically, we learn a bilateral grid $\bar{A}_c^i = B_c(u_s^i) \in \mathbb{R}^{\text{s}\times\text{s}\times\text{d}\times 3\times 4}$ in a lower resolution, and a single-channel guidance map $g_c^i = G_c(I_s^i) \in \mathbb{R}^{W\times H\times 1}$ in full-resolution. We fix $s = 8$ and $d = 8$ in our experiments. B_c and G_c are the trainable networks for estimating the grid and guidance map. Finally, A_c^i is tri-linearly sampled from $\bar{A}_c^i$ using the normalized triplet $(x, y, g_c^i(p))$.

Spatial Transformation (ST). We learn the spatial warping offset $A_s^i = [A_{sx}^i \quad A_{sy}^i] \in \mathbb{R}^{W\times H\times 2}$ along the horizontal and vertical axes. To better preserve the rigidity of the image contents inside hole region, we propose to learn the warping field $\bar{A}_s^i = B_s(u_s^i) \in \mathbb{R}^{\text{s}\times\text{s}\times 2}$ in a lower resolution, and up-sample it to A_s^i using bi-linear interpolation. Finally, $\hat{I}_s^i = \text{Warp}(I_{sc}^i; A_s^i)$. The objective loss to learn the CST module is defined by,

$$\mathcal{L}_{CS}^i = ||M_v \odot M \odot (I_t - \hat{I}_s^i)||_1, \tag{1}$$

where $M_v = \mathbb{1}(I_s^i > 0)$ is the valid mask indicating the pixel regions after initial homography warping.

3.3. Single-Proposal Fusion (SPF) Module

The Single-Proposal Fusion (SPF) module learns to estimate a confidence map and other features for the refined results $\hat{I}_s^i$ from the CST module by merging it with the outputs of a well-trained single image inpainting model called ProFill [69]. The inpainting results from ProFill often generate good structures, so the intuition for the SPF module is that we independently do an image comparison of each proposal against this ProFill reference, to better constrain the overall learning task and learn confidence and difference features that can help the harder downstream multi-proposal fusion task. As shown in Figure 5, the module takes I_t^M, M, a single $\hat{I}_s^i$ and I_g as inputs, where I_g is the output from a single image inpainting method. In this paper, we use a pre-trained ProFill [69] model and freeze its weights while training the whole pipeline. The module outputs a confidence map of $\hat{I}_s^i$ denoted by c_i of the same spatial size as I_t^M. The output $\tilde{I}_s^i$ of merging is

$$\tilde{I}_s^i = c_i \odot \hat{I}_s^i + (1 - c_i) \odot I_g, \tag{2}$$

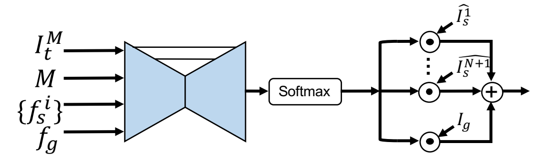

Figure 6: Structure of the Multi-Proposal Fusion (MPF) Module. We feed the UNet with packed features f_s^i and f_g from the SPF module, and learn a spatially-varying merging mask for all the proposals.

The values in the confidence map range from zero to one, and higher-valued regions should contain more informative and realistic pixels. The composited result of merging $I_t^M + M \odot \tilde{I}_s^i$ can also be displayed to the user as an intermediate result demonstrating the performance of a single proposal. In our experiments, we will show that compared to learning to merge multiple $\hat{I}_s^i$ directly, it is better to condition on the outputs from a well-learned SPF module.

Additionally, we utilize a shallow convolutional module to concatenate the learned confidence map c_i and the output of the CST module $\hat{I}_s^i$, and output a three-channel feature map f_s^i to be fed into the final multi-proposal fusion module in section 3.4. Similarly, when we input I_g to the SPF, we obtain the feature f_g. The objective function for learning the SPF is defined as,

$$\mathcal{L}_E^i = ||M \odot (I_t - \tilde{I}_s^i)||_1, \tag{3}$$

and an additional Total Variance loss is imposed on c_i to enforce the smoothness of the map.

$$\mathcal{L}_c^i = \mathcal{L}_{\text{TV}}(c_i), \mathcal{L}_{\text{TV}}(u) = \left\|\frac{\partial u}{\partial x}\right\|_1 + \left\|\frac{\partial u}{\partial y}\right\|_1 \tag{4}$$

3.4. Multi-Proposal Fusion (MPF) Module

The Multi-Proposal Fusion (MPF) module merges the $N+1$ proposals of the refined source images $\hat{I}_s^i$ and the single-image inpainting results I_g together. The module is fed with the packed features f_s^i and f_g from the SPF module. Pixel-wise merging weights $\overline{c_i}, i \in [1, N+1]$ and c_g are learned through a UNet [44] with softmax ($c_g + \sum_{i=1}^{N+1} \overline{c_i} = 1$) by merging different portions of proposals as I_m,

$$I_m = c_g \odot I_g + \sum_{i=1}^{N+1} \overline{c_i} \odot \hat{I}_s^i, \tag{5}$$

Then the final result $I_o = I_t^M + M \odot I_m$ is learned by the objective functions,

$$\mathcal{L}_o = ||M \odot (I_t - I_o)||_1 + VGG(M \odot I_t, M \odot I_o), \tag{6}$$

where the VGG loss matches features of the pool5 layer of a pre-trained VGG19 [46]. Similarly, total variance losses are imposed to the weighting maps $\overline{c_i}$ and c_g, so we have the losses $\mathcal{L}_{\bar{c}}^i = \mathcal{L}_{\mathrm{TV}}(\bar{c})$ and $\mathcal{L}_c^g = \mathcal{L}_{\mathrm{TV}}(c_g)$. Therefore, the overall loss function with $\lambda_1 = 1, \lambda_2 = 1$ becomes

$$\mathcal{L}_{all} = \mathcal{L}_o + \lambda_1 \mathcal{L}_c^g + \textstyle\sum_{i=1}^{N+1} (\mathcal{L}_{CS}^i + \mathcal{L}_E^i + \lambda_2(\mathcal{L}_c^i + \mathcal{L}_{\bar{c}}^i)). \quad (7)$$

4. Experimental Results

4.1. Datasets and Implementation

Datasets. We trained the model on the RealEstate10K dataset [75]. This was collected from YouTube videos labelled as real estate footage. In total it consists of more than 8000 video clips with length from 1 to 10 seconds. For each clip, we randomly sampled pairs of images with a displacement of 10, 20, and 30 frames apart. We call this "Frame Displacement" (FD). This resulted in 188184 frame pairs for training, and 20290 pairs for testing. We generated random free-form brush-and-stroke holes like in DeepFillv2 [67]. We also collected 3K more pairs of real user-provided image pairs to serve as practical user cases for testing.

For training the Color-Spatial Transformer (CST), although RealEstate10K contains sufficient samples with real multi-view data and different exposures across image pairs, it lacks image pairs with large color inconsistency. Therefore, we synthesized misaligned color-different images from the MIT-Adobe5K dataset [5], and uniformly mixed these data with RealEstate10K for training. Adobe5K contains 5000 images, and for each image it provides five additional expert-retouched images to form 5000 sets in total. We regard the original samples as target images and synthesized the misaligned source images using the method in [7]. We make two binary variables for whether there is a color difference (C) and whether there is spatial misalignment (S), and synthesized pairs with CS, $C\bar{S}$, $\bar{C}S$ and $\bar{C}\bar{S}$ with equal probability from 4000 sets to form a balanced training set, leaving 1000 sets for validation.

Implementations. We obtained a pre-trained OANet model for image feature matching and outlier rejection. We applied the pretrained model of Hu *et al*. [16] to estimate the depth map from a single target image. We also obtained a pre-trained ProFill [69] from the authors. All the above-mentioned model weights were frozen during training. Additionally, we pre-trained the CST module using the mixed dataset in advance for 400 epochs, and froze its weights afterwards. Finally, the whole pipeline was trained end-to-end for 400 more epochs. We used a patch size of 256×256 for training and arbitrary size for inference, and a learning rate of 10^{-4} with decay rate 0.5 after 200 epochs. We used the Adam optimizer [22] with betas (0.9, 0.999). The code is implemented in PyTorch [38].

4.2. Baseline Models

We chose baselines that are similar to, but may not exactly the same as our task, including approaches addressing image stitching [68], optical flow-guided video inpainting [61], non-local patch matching for multiple photo inpainting [37], and a state-of-the-art single image inpainting method [69] with the reference image concatenated so the method has access to the same inputs as the rest.

APAP [68]: As-Projective-As-Possible is a baseline image stitching algorithm that resolves depth parallax. We used the official Matlab [1] implementation for testing.

DFG [61]: Deep Flow-Guided Video Inpainting treats video inpainting as pixel propagation. It fills the holes by completing the optical flow field estimated by FlowNet2.0 [18]. We used their official[2] pre-trained model for testing.

OPN [37]: Onion-Peel Network is a recent work addressing video and group photo inpainting using non-local attention blocks. We used their official PyTorch code[3].

ProFill [69]: ProFill is a state-of-the-art single-image inpainting method that also contains a contextual attention module [66]. We used the official pre-trained model [4] from the authors. When testing, we fed in the target with the homography-warped source image. Before testing on RealEstate10K, we also fine-tuned OPN and ProFill on RealEstate10K training frames for fairness.

4.3. Qualitative Comparison

Results on User-Provided Images. In Figure 7, we show visual results of testing on real user-provided images. We indicate the hole region on the target image, and crop only the region of interest due to the space limits. More results can be found in the supplementary material. APAP and DFG well-preserve the source image contents due to the global homography warping, but they still suffer from color inconsistencies and alignment issues. We also experimented with combining Poisson blending with APAP but found it gives color bleeding artifacts: see the supplemental for details. OPN usually works well when there are multiple reference frames which have similar scales and color distributions within the same video clips. However, if only one source reference image exists, the non-local attention module struggles to search for similar local patches and fails. ProFill with the contextual attention module usually does well in searching for textures, but the estimated intermediate coarse results cannot be matched with specific image contents. Thus the reference-based ProFill can only achieve texture or object removal but not background contents recovery. Compared to them, ours better reuses the background patterns and achieves a content-aware alignment and

[1] APAP: https://cs.adelaide.edu.au/∼tjchin/apap/
[2] DFG: https://github.com/nbei/Deep-Flow-Guided-Video-Inpainting
[3] OPN: https://github.com/seoungwugoh/opn-demo
[4] ProFill: https://zengxianyu.github.io/iic/

Figure 7: Comparison with baselines on challenging user-provided image pairs. For better visualization, we only crop the regions of interest from the whole target and source images. Please zoom in to see the details.

composition. The generated results are more faithful to and compatible with the target image. The multi-homography proposal approach provides more options for warping initialization when the matched features are too complex for a single homography. It helps to resolve challenging cases when the hole regions do not belong to the dominant plane in the image as shown in row four. In Figure 8 we show intermediate results indicating regions selected to form the final results and how the CST resolves misalignment.

4.4. Quantitative Comparison

Results on RealEstate10K. The quantitative comparison on RealEstate10K is shown in Table 1. OPN and ProFill are more suitable for large batch testing. We tested them on the entire testing set. Results on cropped image pairs with Frame Displacement (FD) 10, 20 and 30 are reported in terms of PSNR, SSIM and LPIPS scores [72] based on AlexNet [23]. APAP and DFG are not suitable for large batch testing and their performance may be influenced by non-existing regions, so we sampled a 300-image subset from FD=10 as *Small Set* to test. Results showed that contextual-attention based ProFill failed to faithfully reconstruct the source contents. Optical-flow based DFG achieved better results by smoothly completing the flow field. OPN with atomic patch matching was not better than our warping-based approach. The TransFill thus demonstrated its superiority in faithful reconstruction.

User Study on User-Provided Images. To better evaluate the performance on our user-provided images, we conducted a user study via Amazon Mechanical Turk (AMT). We compared our method with each baseline separately and presented users with binary choice questions. We requested the users to choose one fill result which looks more realistic and faithful. To ensure the reliability, we used a pre-qualification test as well as check questions, as we explain in the supplementary material. For each method pair, we randomly sampled 80 examples, and each example was evaluated by 7 independent users. For each sample, one method was regarded as "preferred" if at least 5 users selected it. Samples voted by 3 or 4 users are considered confusing samples and filtered out. We reported TransFill's Preference Rate (PR) in Table 1. The high preference rate demonstrates the effectiveness of TransFill. We also conducted a one-sample permutation t-test with 10^6 samples by assuming a null hypothesis that on average 3.5 users prefer one method. The p-values are all sufficiently small so we can draw the conclusion that the preference for our method was statistically significant.

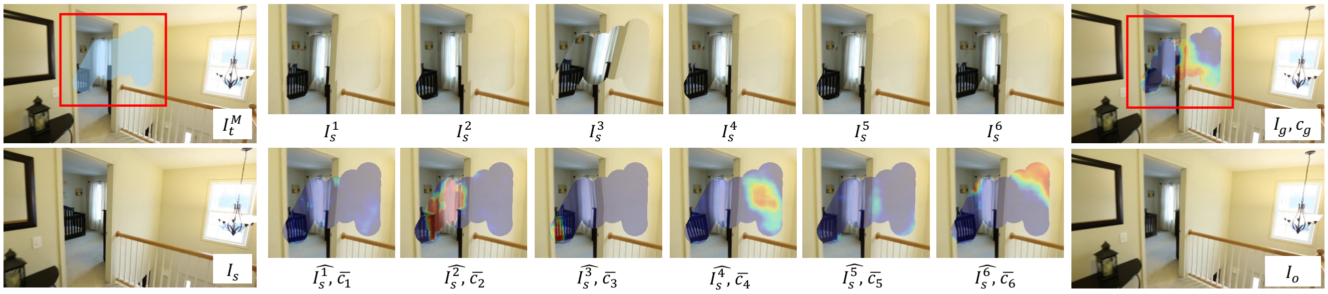

Figure 8: Visualization of intermediate results. I_s^i are the initialized homography warping of the source image I_s. $\hat{I}_s^i$ are the learned spatial and color transformation of I_s^i. The final result I_o is the merging of $\hat{I}_s^i$ and the results of ProFill I_g by pixel-wise weights $\overline{c_i}$ and c_g overlaying on the images. The result draws from regions that are better aligned: on the left from $\hat{I}_s^2$, in the center from the single image inpainting I_g, and on the right from $\hat{I}_s^4$ and $\hat{I}_s^6$. Zoom in for better visualization.

Table 1: Quantitative Comparisons and User Study. **FD**: Frame Displacement. **PR**: Preference Rate

	RealEstate10K: PSNR ↑/ SSIM ↑ / LPIPS ↓					User-provided Images: User Study	
Model	FD=10	FD=20	FD=30	All	Small Set	PR	p-value
APAP [68]	-	-	-	-	31.94 / 0.9738 / 0.0251	90.76%	$p < 10^{-6}$
DFG [61]	-	-	-	-	36.17 / 0.9873 / 0.0155	87.50%	$p < 10^{-6}$
OPN [37]	33.45 / 0.9765 / 0.0201	32.47 / 0.9734 / 0.0258	31.32 / 0.9699 / 0.0320	32.43 / 0.9734 / 0.0261	33.40 / 0.9771 / 0.0207	95.65%	$p < 10^{-6}$
ProFill [69]	31.18 / 0.9689 / 0.0423	31.14 / 0.9687 / 0.0425	30.83 / 0.9683 / 0.0440	31.05 / 0.9687 / 0.0429	30.95 / 0.9690 / 0.0419	81.67%	$p < 10^{-6}$
TransFill (Ours)	**39.59 / 0.9919 / 0.0116**	**37.39 / 0.9877 / 0.0162**	**35.62 / 0.9839 / 0.0213**	**37.58 / 0.9879 / 0.0164**	**38.83 / 0.9914 / 0.0126**	-	-

Table 2: Ablation Study on Multi-Homography Proposals.

Clustering	N	Outlier Rejection	PSNR↑	SSIM↑	LPIPS↓
Depth	N=5	OANet	**37.576**	**0.9879**	0.0164
Depth	N=5	Ratio Test [32]	37.444	0.9876	0.0168
Random	N=5	OANet	37.499	0.9873	0.0166
Spatial	N=5	OANet	37.384	0.9876	0.0169
Depth	N=3	OANet	37.537	0.9878	**0.0162**
None	N=1	OANet	37.092	0.9868	0.0172

Table 3: Color-Spatial Transformation. **C**: Color, **S**:Spatial

Order	PSNR↑	SSIM↑	LPIPS↓
$C \rightarrow S$	**37.576**	**0.9879**	0.0164
$S \rightarrow C$	37.566	**0.9879**	**0.0163**
Only S	36.717	0.9866	0.0182
Only C	36.228	0.9849	0.0179

4.5. Ablation Study

Type and Number of Multi-Homography Proposals. This ablation study was conducted on the testing set of the RealEstate10K. For each alternative, we re-trained the model. We compared the proposed depth-based points clustering methods with other alternatives including random and spatial clustering in Table 2. When we proposed five homography matrices, depth-based clustering works best. The results were fairly close when we set N to either 3 or 5, but $N = 5$ was slightly better in PSNR. However, these are much better than using just one global homography.

Color-Spatial Transformation Module. Table 3 shows that the order of the Color-Spatial Transformer did not make too much difference. However, according to our experiments, adjusting the color first made the training converge faster since the guidance map was computed from a fixed I_s^i. Table 3 also demonstrated that both the Color and Spatial Transformer were necessary.

Pipeline Components. Table 4 indicates that refining the source image with CST outperforms directly merging the initialized homography-warped images. SPF and its output confidence c_i effectively guided the learning of MPF. The proposed full pipeline achieved the best performance.

Table 4: Ablation Study on Pipeline Components. **CST**: Color-Spatial Transformer, **SPF**: Single-Proposal Fusion.

CST	SPF	PSNR↑	SSIM↑	LPIPS↓
✓	✓	**37.576**	**0.9879**	**0.0164**
✗	✓	35.579	0.9838	0.0183
✓	✗	36.710	0.9861	0.0188
✗	✗	33.484	0.9782	0.0249

5. Limitations, Discussion and Conclusions

The proposed method has limitations in certain situations. First, the pipeline may not work well on extreme low-light or texture inputs containing very few SIFT feature points. Second, our homography-based transformation is not suited for image pairs with extreme viewpoint changes. Third, the current model may struggle to transfer color if the lighting environment is very different, such as day to night. This is because we use an effective bilateral grid color matching, but do not incorporate any specialized models that reason further about lighting (e.g. [11]). Additionally, we utilize the pre-trained ProFill to fill the missing pixels so the final generation quality highly depends on the performance of the single image inpainting module ProFill. That module could be replaced by other state-of-the-art models, and could potentially be optimized with the multi-fusion pipeline together. We leave that for future work.

In conclusion, we contribute a multi-source image inpainting model based on multiple homography, deep warping and color harmonization. The results outperform state-of-the-art single image and multi-source inpainting methods, especially when the hole contains complicated depth.

References

[1] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen. Interactive digital photomontage. In *ACM SIGGRAPH 2004 Papers*, pages 294–302. 2004. 3

[2] Coloma Ballester, Marcelo Bertalmio, Vicent Caselles, Guillermo Sapiro, and Joan Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE transactions on image processing*, 10(8):1200–1211, 2001. 2

[3] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24, 2009. 1, 2

[4] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424, 2000. 2

[5] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 6

[6] Wenyan Cong, Jianfu Zhang, Li Niu, Liu Liu, Zhixin Ling, Weiyuan Li, and Liqing Zhang. Dovenet: Deep image harmonization via domain verification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8394–8403, 2020. 3

[7] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016. 3, 6

[8] Martin A Fischler and Robert C Bolles. 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. 4

[9] John Flynn, Ivan Neulander, James Philbin, and Noah Snavely. Deepstereo: Learning to predict new views from the world's imagery. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5515–5524, 2016. 2

[10] Junhong Gao, Seon Joo Kim, and Michael S Brown. Constructing image panoramas using dual-homography warping. In *CVPR 2011*, pages 49–56. IEEE, 2011. 3

[11] Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. Fast spatially-varying indoor lighting estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6908–6917, 2019. 8

[12] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017. 3, 4, 5

[13] Miguel Granados, Kwang In Kim, James Tompkin, Jan Kautz, and Christian Theobalt. Background inpainting for videos with dynamic objects and a free-moving camera. In *European Conference on Computer Vision*, pages 682–695. Springer, 2012. 2

[14] Charles Herrmann, Chen Wang, Richard Strong Bowen, Emil Keyder, Michael Krainin, Ce Liu, and Ramin Zabih. Robust image stitching with multiple registrations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 53–67, 2018. 3

[15] Berthold KP Horn and Brian G. Schunck. " determining optical flow": A retrospective. 1993. 3

[16] Junjie Hu, Mete Ozay, Yan Zhang, and Takayuki Okatani. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. 2019. 4, 6

[17] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017. 2

[18] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 3, 6

[19] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 4

[20] Jiaya Jia, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum. Drag-and-drop pasting. *ACM Transactions on graphics (TOG)*, 25(3):631–637, 2006. 3

[21] Stephen C Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967. 4

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 7

[24] Kyu-Yul Lee and Jae-Young Sim. Warping residual based image stitching for large parallax. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8198–8206, 2020. 3

[25] Sungho Lee, Seoung Wug Oh, DaeYeun Won, and Seon Joo Kim. Copy-and-paste networks for deep video inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4413–4421, 2019. 3

[26] Shiwei Li, Lu Yuan, Jian Sun, and Long Quan. Dual-feature warping-based motion model estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4283–4291, 2015. 3

[27] Liang Liao, Jing Xiao, Zheng Wang, Chia-wen Lin, and Shin'ichi Satoh. Guidance and evaluation: Semantic-aware image inpainting for mixed scenes. *arXiv preprint arXiv:2003.06877*, 2020. 2

[28] Feng Liu, Michael Gleicher, Hailin Jin, and Aseem Agarwala. Content-preserving warps for 3d video stabilization. *ACM Transactions on Graphics (TOG)*, 28(3):1–9, 2009. 3

[29] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018. 2

[30] Shuaicheng Liu, Ping Tan, Lu Yuan, Jian Sun, and Bing Zeng. Meshflow: Minimum latency online video stabilization. In *European Conference on Computer Vision*, pages 800–815. Springer, 2016. 3

[31] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999. 3

[32] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 4, 8

[33] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 2

[34] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. 2019. 2

[35] Alasdair Newson, Andrés Almansa, Matthieu Fradet, Yann Gousseau, and Patrick Pérez. Video inpainting of complex scenes. *Siam journal on imaging sciences*, 7(4):1993–2019, 2014. 2

[36] Ty Nguyen, Steven W Chen, Shreyas S Shivakumar, Camillo Jose Taylor, and Vijay Kumar. Unsupervised deep homography: A fast and robust homography estimation model. *IEEE Robotics and Automation Letters*, 3(3):2346–2353, 2018. 3

[37] Seoung Wug Oh, Sungho Lee, Joon-Young Lee, and Seon Joo Kim. Onion-peel networks for deep video completion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4403–4412, 2019. 1, 3, 6, 8

[38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. 6

[39] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016. 2

[40] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, pages 313–318. 2003. 3

[41] Francois Pitie, Anil C Kokaram, and Rozenn Dahyot. N-dimensional probability density function transfer and its application to color transfer. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1434–1439. IEEE, 2005. 3

[42] Erik Reinhard, Michael Adhikhmin, Bruce Gooch, and Peter Shirley. Color transfer between images. *IEEE Computer graphics and applications*, 21(5):34–41, 2001. 3

[43] Yurui Ren, Xiaoming Yu, Ruonan Zhang, Thomas H. Li, Shan Liu, and Ge Li. Structureflow: Image inpainting via structure-aware appearance flow. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[44] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 5

[45] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4938–4947, 2020. 3

[46] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[47] Yuhang Song, Chao Yang, Yeji Shen, Peng Wang, Qin Huang, and C-C Jay Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356*, 2018. 2

[48] Deqing Sun, Stefan Roth, and Michael J Black. Secrets of optical flow estimation and their principles. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 2432–2439. IEEE, 2010. 3

[49] 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 conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 3

[50] Kalyan Sunkavalli, Micah K Johnson, Wojciech Matusik, and Hanspeter Pfister. Multi-scale image harmonization. *ACM Transactions on Graphics (TOG)*, 29(4):1–10, 2010. 3

[51] Michael W Tao, Micah K Johnson, and Sylvain Paris. Error-tolerant image compositing. In *European Conference on Computer Vision*, pages 31–44. Springer, 2010. 3

[52] Yi-Hsuan Tsai, Xiaohui Shen, Zhe Lin, Kalyan Sunkavalli, Xin Lu, and Ming-Hsuan Yang. Deep image harmonization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3789–3797, 2017. 3

[53] Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 551–560, 2020. 2

[54] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. Underexposed photo enhancement using deep illumination estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6849–6857, 2019. 3

[55] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1385–1392, 2013. 3

[56] Yonatan Wexler, Eli Shechtman, and Michal Irani. Space-time completion of video. *IEEE Transactions on pattern analysis and machine intelligence*, 29(3):463–476, 2007. 2

[57] Oliver Whyte, Josef Sivic, and Andrew Zisserman. Get out of my picture! internet-based inpainting. In *Proceedings of the 20th British Machine Vision Conference, London*, 2009. 2

[58] Jonas Wulff and Michael J Black. Efficient sparse-to-dense optical flow estimation using a learned basis and layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 120–130, 2015. 3

[59] Cun Xiaodong and Pun Chi-Man. Improving the harmony of the composite image by spatial-separated attention module. *arXiv preprint arXiv:1907.06406*, 2019. 3

[60] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. Foreground-aware image inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5840–5848, 2019. 1

[61] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep flow-guided video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2019. 3, 6, 8

[62] Tianfan Xue, Michael Rubinstein, Ce Liu, and William T Freeman. A computational approach for obstruction-free photography. *ACM Transactions on Graphics (TOG)*, 34(4):1–11, 2015. 3

[63] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6721–6729, 2017. 2

[64] Nianjin Ye, Chuan Wang, Shuaicheng Liu, Lanpeng Jia, Jue Wang, and Yongqing Cui. Deepmeshflow: Content adaptive mesh deformation for robust image registration. *arXiv preprint arXiv:1912.05131*, 2019. 3

[65] Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7508–7517, 2020. 2

[66] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018. 1, 6

[67] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4471–4480, 2019. 1, 2, 6

[68] Julio Zaragoza, Tat-Jun Chin, Michael S Brown, and David Suter. As-projective-as-possible image stitching with moving dlt. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2339–2346, 2013. 6, 8

[69] Yu Zeng, Zhe Lin, Jimei Yang, Jianming Zhang, Eli Shechtman, and Huchuan Lu. High-resolution image inpainting with iterative confidence feedback and guided upsampling. *arXiv preprint arXiv:2005.11742*, 2020. 2, 5, 6, 8

[70] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Long Quan, and Hongen Liao. Learning two-view correspondences and geometry using order-aware network. *International Conference on Computer Vision (ICCV)*, 2019. 3, 4

[71] Jirong Zhang, Chuan Wang, Shuaicheng Liu, Lanpeng Jia, Nianjin Ye, Jue Wang, Ji Zhou, and Jian Sun. Content-aware unsupervised deep homography estimation. *arXiv preprint arXiv:1909.05983*, 2019. 3

[72] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 7

[73] Yinan Zhao, Brian Price, Scott Cohen, and Danna Gurari. Guided image inpainting: Replacing an image region by pulling content from another image. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1514–1523. IEEE, 2019. 3

[74] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1438–1447, 2019. 2

[75] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. 2, 4, 6

[76] Jun-Yan Zhu, Philipp Krahenbuhl, Eli Shechtman, and Alexei A Efros. Learning a discriminative model for the perception of realism in composite images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3943–3951, 2015. 3