Inpainting of Wide-baseline Multiple Viewpoint Video

Andrew Gilbert, Matthew Trumble, Adrian Hilton, *Member, IEEE* and John Collomosse *Member, IEEE*,

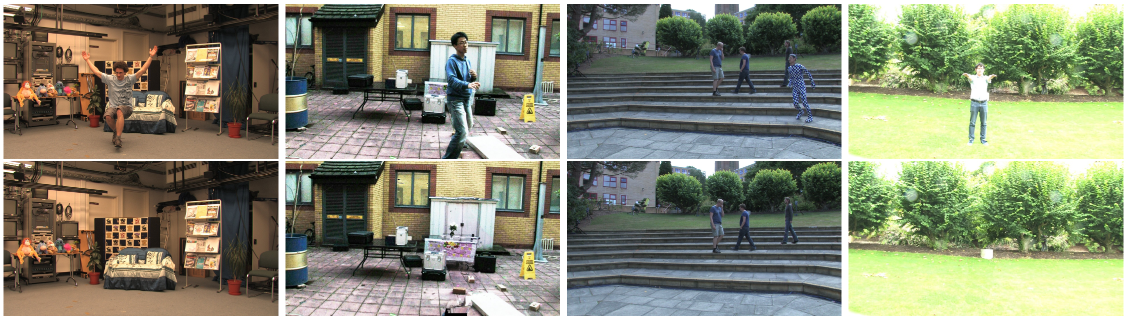


Fig. 1: Our proposed method in-paints regions in the principal (hero) camera view (top), using texture from multiple cameras across a wide baseline. The resulting video frames (bottom) are completed using texture unavailable in the principal view. Our approach handles moving camera, clutter and occlusion.

Abstract—We describe a non-parametric algorithm for multiple-viewpoint video inpainting. Uniquely, our algorithm addresses the domain of wide baseline multiple-viewpoint video (MVV) with no temporal look-ahead in near real time speed. A Dictionary of Patches (DoP) is built using multi-resolution texture patches reprojected from geometric proxies available in the alternate views. We dynamically update the DoP over time, and a Markov Random Field optimisation over depth and appearance is used to resolve and align a selection of multiple candidates for a given patch, this ensures the inpainting of large regions in a plausible manner conserving both spatial and temporal coherence. We demonstrate the removal of large objects (e.g. people) on challenging indoor and outdoor MVV exhibiting cluttered, dynamic backgrounds and moving cameras.

Index Terms—Computer Graphics, Inpainting, Multiple Viewpoint Video

1 Introduction

Multiple viewpoint video is becoming commonplace in film and music video production, where shots are captured using multiple, wide-spaced, synchronised cameras for later editing. Frequently it is desirable to edit such footage to remove objects; e.g., to remove an actor in a motion-capture suit to be later replaced with an animated virtual character.

There is increasing work in image inpainting, including commercial inpainting products such as Adobe Photoshop's Content Aware fill [2]. However, there is far less progress for video inpainting, which requires the copying or hallucination of texture to fill the removed region and is currently manually performed through rotoscoping to create per-frame 'clean plates'. Producing even short sequences under this work-flow can require many person-months.

This paper presents the *first video inpainting algorithm for wide-baseline multiple viewpoint videos (MVV)*. We assume a scene observed by multiple static (witness) cameras and a moving or static principal (hero) camera. The goal is to inpaint the unwanted object in the principal view, using texture synthesised from any combination of the principal or witness views at a frame a second, orders of magnitude fast than other state of the art approaches, which can take around 1500 to 4000 seconds a frame. The texture should be visually plausible and exhibit coherence both spatially (absent of visual discontinuities) and temporally (without flicker).

All authors are with the Centre for Vision Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK e-mail: a.gilbert@surrey.ac.uk. J.Collomosse also at Creative Intelligence Lab, Adobe Research USA

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org.
Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx/

Our proposed approach can *peek* behind large object occlusions, to inpaint the scene even though the background is never revealed to the principal camera. Figure 1 illustrates our inpainting results; the edge of the sofa and white box are never revealed in the video to the principal camera.

Our core technical contribution is an optimisation framework through which appropriate texture, sampled from a set of patches collated in a Dictionary of Patches (DoP) from multiple available viewpoints, then selected and combined to complete the missing region. The patches are optimised over time by several factors including spatiotemporal coherence, patch depth, view coherence, age, and historic popularity of each patch. These constraints are used to determine a globally optimal patch set for inpainting. Optimisation adopts a multi-resolution strategy to expedite inpainting further enhancing accuracy with a coarse to fine alignment step.

2 Related Work

Texture synthesis for image inpainting has been extensively studied. Initially, approaches focused upon greedy per-pixel iterative methods [5], later extended to global optimisation by posing image completion as a pixel-labelling problem [24]. Recently due to their ease of parallelism, per pixel sampling and filtering has been re-explored [23]. They use the redundancy of multiple frames in a video to inpaint areas. Improved retention of the local structure can be achieved by seamlessly quilting small patches sampled from elsewhere in the same [9, 10] or visually similar images [15]. The re-projection of the 2D patches into 3D [11] allows for more accurate correlation, we employ this principle of re-projection for our MVV approach. Layered data driven patch-based synthesis of Lightfield data has been examined by Zhang [38], decomposing the central view into different depth layers, and presenting it to the user for specifying the editing goals.

Patch based image inpainting has been successfully extended to monocular video inpainting. A direct extension of patches to small space-time cuboids was explored by Wexler [37] adapting the greedy iterative completion of [10], similarly enforcing local coherence through consideration of adjacent (spatiotemporal) patches. The technique is sensitive to fast object motion. However, Newson extended and robustified the approach [27] providing the current state of the art in video inpainting. Like this approach, they use a shift-map principle to find the most coherent patches in space and time. However we use the reconstructed geometric proxy based witness views to sample additional patches from, and we also incorporate optimisation terms related to the patch age, depth and freshness, in particular, to ensure recent coherent patches are incorporated. A generalisation to simple camera motion was proposed through foreground segmentation in [28]. To deal with circumstances in which background is not observed unoccluded during the sequence, additional constraining assumptions on the motion of the segmented foreground object are introduced, e.g. cyclic motion [33]. While Huang [18] proposed to inpaint the regions guided by constraints on planar structure, and Darabi [8] incorporated geometric transformation of the patches. Extending this Zhu [39] used similar images from the internet to provide the source data, and inpaint through point matching and warping of the images. Indicating the importance of compensating for the appearance variation due to perspective distortion from wide baseline cameras. Extending the per-pixel inpainting approaches, a space-time extension of shift-maps [30] was proposed in [13] aligning the video frames via homographies and to defining the occlusion-aware per-pixel flow. The method is semi-automatic requiring user-input to determine occlusion regions and uses multi-resolution matching over space-time to reduce the complexity of the video inpainting. While there is also a similarity with monocular HDR video reconstruction, infilling information from previous frames through modelling pixel-level optical flow [22] or temporally and spatially constrained patch based synthesis [21]. While Huang [19] jointly estimated optical flow and colour in the missing region to synthesise missing regions in videos in a temporally coherent fashion.

Multiple viewpoint video (MVV) inpainting has received little attention. Literature addresses only the narrow baseline case, typically that of stereo pairs [16,25,32,34,35,40]. Given the small extent of the area to be filled, approaches approximate missing content by extrapolating adjacent colour-depth information and performing patch matching over the extrapolated pixels.

Thonat *et al.* [31] use structure from motion to re-project the images and a coarse to fine patch based optimiser - the integration of patch data over time to enhance temporal coherence is not possible as with our weighted dictionary of patches. While in image synthesis, Hu [17] proposed PatchNet, a patch-based image representation for Interactive Image Editing. PatchTable [4], proposed an index data structure to accelerate the access and representation the patches. It performs offline precomputation (optimisation through hashing and precomputed neighbour mapping), however, we obtain the efficiency through online optimisation of the patches, intelligently culling patches that are unused or irrelevant to the inpainting challenge.

3 Multi-viewpoint Video InPainting

Fig. 2 provides an overview of the proposed wide base-line MVV inpainting method. A coarse multi-planar model of the scene geometry is computed via Delaunay triangulation over recovered 3D positions of sparse feature points matched between views (Fig. 2a). This geometric proxy is a basis for reprojecting texture from the witness views to the principal view containing the region to be inpainted (Fig. 2b). A regular grid of overlapping patches is constructed over the area to be filled in the principal view. To fill this grid, a dictionary of patches (DoP) comprising of

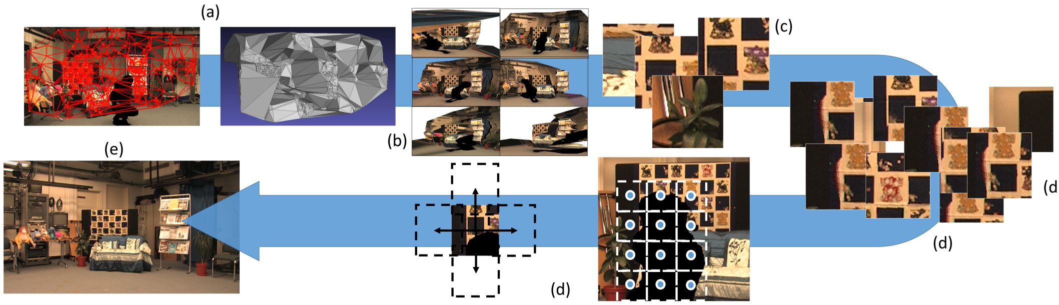


Fig. 2: Overview of the inpainting pipeline

sampled candidates from the principal and witness views is used (Fig. 2c) The patch consists of visual (RGB pixel) and generated depth information in x,y. The DoP is updated over time and contains patches from previous frames. Interpreting the grid as a graph, where patches (nodes) are connected in a lattice by adjacency, we use a Markov Random Field (MRF) to assign patches from the DoP to the grid. Assignments are made to encourage spatiotemporal coherence, visual plausibility, and patch freshness (Fig. 2d). Once converged, the region is inpainted with the assigned patches (Fig. 2e).

3.1 In-painting Formulation

We consider a set of images $\mathscr{I}_t = \{I_1, \ldots I_n\}$, with overlapping viewpoints comprising a single frame of an MVV sequence, where the principal or *hero* view I_h contains a mask region Ω_h for inpainting. Each frame I_t is processed in turn, replacing Ω_h using small patches from the image of texture of size $h \times w$, sampled from any valid region of any viewpoint $I_c \setminus \Omega_c c \in \mathscr{I}_{1..t}$.

We propose a global optimisation for inpainting Ω_h, finding the optimal combination of 2D visual texture patches These patches represent vertices $i \in \mathscr{V}$ within a discrete random field consisting of an undirected graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ while the edges $\mathscr{E}$ connecting neighbouring spatial vertices i, j within the same time-step t and the neighbouring vertex at $t-1$. Observing prior patch-based inpainting work [9,10] and image statistics [12,29], we assume the choice of patch for p is dependent in its immediate neighbours. The optimal inpainting of Ω_h is the label map x minimizing the energy $E(x) = D(x) + C(x)$, between $D(x)$ the single vertex data potential, and $C(x)$ the spatio-temporal coherence i.e pairwise potential. We unpack this energy term in detail within Sec. 3.4 (c.f. eq. 1).

3.2 Handling Dynamic Patches with depth

Patches of dynamic nonrigid objects such as people within the scene are smaller than the static background. Therefore, the MRF optimisation can often seek to ignore them incorrectly for the greater global static background. We propose to append an approximate depth map into the RGB image to form a 4D image to incorporate and promote moving objects. The geometric proxy from the Delaunay triangulation can be converted into an approximate depth map by using the world 3D coordinates from the key points. The depth value for each candidate patch is computed from the nearest keypoint depth to provide an approximate value and visualised as a depth image and appended to the three channel RGB image

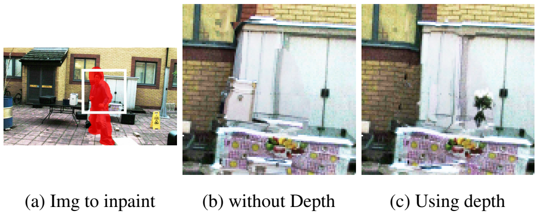
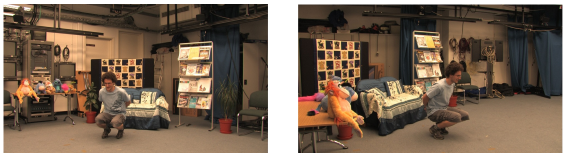

(a) Img to inpaint (b) without Depth (c) Using depth

Fig. 3: A *courtyard* frame, where without depth, the vase is missed, and a silver box incorrectly inpainted.

The depth PVH is essential to distinguish between mid and background objects, but it can also enforce the inpainting of otherwise ignored small static objects. Fig 3b ignores the depth and produces coherent inpainting but missing the vase of flowers. However, Fig 3c considers the depth and correctly inpaints the vase

3.3 Geometric Proxy

If all images in $\mathscr{I}_t$ had similar viewpoints, it would be possible to sample the DoP without little or no need for distortion to correct the change in viewing direction. However, the advantage of wide baseline MVV is the ability to observe background 'behind' the matted object from an alternate view. For example to see behind the person in Fig 4a, patches from the witness views such as Fig 4b, will provide



(a) (b)

Fig. 4: We exploit the benefits of wide-baseline MVV, content masked in (a) can be observed in (b)

the otherwise occluded detail.

Directly sampling rectilinear patches, e.g., of the sofa in Fig. 4a would lead to heavy distortion that would increase $E(x)$. To reduce the energy and improve visual plausibility, we warp to correct for foreshortening when sampling texture from witness cameras. We recover a rough low-resolution proxy of the scene using sparse wide baseline stereo [20]. The triangulation of sparse correspondences derived from key-points detected in two or more views produces a depth estimate to generate a sparse 3D point cloud. Resulting in a sparse set of 3D points which to be readily converted into a mesh through Delaunay triangulation the result is a view-dependent 3D reconstruction or ‘geometric proxy’

For the k^{th} image view, projective texture mapping projects the image I_k onto the geometric proxy; per-view 3D mesh, this is then reprojected to the viewpoint of I_h obtaining the reprojected transformed colour image I_k^h in the h^{th} or principal image view.

3.4 MRF Optimisation

The overall process of inpainting the video is illustrated in Algorithm 1.

Given a set of all grid vertices $\mathscr{V} = \{i_1, ..., i_n\}$ and $p_i \in \mathscr{P}$, where p_i denotes the patch label associated with the i^{th} patch, and f_i is the frame the patch was sampled from, to favour newer patches. The optimisation minimizes an energy function $E(x) = D(x) + C(x)$ that balances the data term $D(x)$ comprising the visual plausibility ψ_s of the 4D video + depth patch weighted by the freshness of the patch ω_i and $C(x)$ the pairwise spatio-temporal smoothness coherence ψ_{ij} weighted by the camera distance $\gamma_{i,j}$ between the patches with the labels p_i and p_j:

$$E(x) = \sum_{i \in \mathscr{V}} \omega_i \psi_s(p_i) + \frac{1}{|N_i|} \sum_{i \in \mathscr{V}, j \in N_i} \gamma_{ij} \psi_{ij}(p_i, p_j) \quad (1)$$

Algorithm 1: Inpainting of video, using wide base line MVV

```
Data: Input MVV, mask region $\Omega_h$
Result: Inpainted hero $I_h$ video
Go through the frames in the sequence
for $t$ to $n$ do
    Grid $I_h$ to identify nodes to optimise;
    optimise for each vertex $i$
    for $i \in \mathscr{V}$ do
        1, Compute the freshness of the patch, $\omega_i$;
        2, compute patch appearance and depth
          similarity wrt the label, $\psi_s(p_i)$;
        3, Identify patch coherency with respect to
          spatio-temporal neighbouring patches,
          $\psi_{ij}(p_i, p_j)$;
        4, Maintain the age of the Patch $f_i$ while
          $(t - f_i) < \tau$ do
            Add patch to DoP;
        Optimise E(x) = D(x) + C(x) (Eq. 1);
```

Where N defines the set of the neighbouring patches in space and time, to minimise the energy difference for spatial temporal coherence between the candidate i and the node. We first compute visual plausibility of the 4D RGB + Depth patch to the corresponding grid cell using normalized sum of squared differences (SSD).

$$\psi_s(p_i) = \sum ||\Omega_h \odot (\mathscr{S}(p_i) - I_h)||_2^2 \quad (2)$$

Where p_i is the patch label for i and $\mathscr{S}(.)$ is the function that returns a putative image containing only p_i as non zero pixel wise values, and Ω_h is the mask, whose values are zero in the inpainting target region and one otherwise, combined with element-wise multiplication. This would mean that when the node is completely within the inpainting target region This visually plausible score is weighted by a factor ω_i inversely proportional to the freshness of the patch; a combination of its age and when it was last used to inpaint a frame. Description of this weighting is deferred to subsec. 3.6. In addition to minimising the energy difference between the candidate i and the node, the optimisation considers spatial-temporal coherency with respect to the set of neighbouring patches in space and time. N defines the set of the neighbouring patches in space and time, and the coherence is measured through the sum of square difference ψ_{ij} of the pixel values in the overlap area between neighbouring patches i, j in both time and space.

3.5 Multi-Resolution Patches

For maximal spatial coherence, ideally, the candidate patch would completely intersect with the matte Ω_h. However,

To approximate the desire for a single large candidate patch, a constraint is incorporated into the energy function that favours neighbouring patches from the same view despite any scale changes. A visibility penalty γ is introduced, reflecting a cost to proposing adjacent patches where there is a large difference in viewing angle of cameras from which those patches were sampled. The principal vectors of the camera given the vertices, p_i and p_j are defined as c_i and c_j respectively, therefore

$$\gamma(p_i, p_j) = 1 + \frac{1 + \arccos(c_i \cdot c_j)}{|\mathscr{I}_t|} \tag{3}$$

Under this multi-resolution approach, iterative MRF optimizations (subsec. 3.7) are performed at increasing scale $s = [1...4]$. Fig. 5 demonstrates inpainting as s increases. As the MRF is solved at each scale, the patches of $\mathscr{V}$ selected by the algorithm are composited into the principal view I_h. The solved grid is composited one patch at a time, working from the perimeter of the region Ω_h inwards in a greedy manner. Due to the use of a coarse geometric estimate of the scene, it is possible for the most highly correlated patches to be slightly misaligned with the hero view. Therefore, we propose a recursive coarse to fine patch alignment. Each patch area to be inpainted is subdivided into four sub-regions,

3.6 Dictionary of Patches (DoP)

As each patch is sampled from a reprojected view and entered into the DoP ($\mathscr{V}$), a per-patch frame id f_i is used to monitor the patch *freshness*, ω_i.

$$\omega_i = \frac{\tau + (t - f_i)}{\tau} \tag{4}$$

This adaptive culling of patches allows for a dictionary of historical patches to be maintained and applied to the minimisation framework without optimising over every possible patch and biasing toward the freshest content as this is most likely to reflect the true background.

3.7 Optimization

The overall energy function to be minimized is solved via Iterated Conditional Modes (ICM) [6]. ICM proceeds by choosing an initial random label configuration for the vertices. It iterates over each vertex in the graph and calculates the value that minimises the energy given the current values of all the variables in its neighbourhood in time and space. At the end of an iteration, the new values for each variable become the current values, and the next iteration begins. In general, within MRF optimisation, an important step is label pruning. The extreme situation is that each node examines all possible labels $\mathscr{P}_t$, this is highly inefficient as a majority of labels are ill-suited for a specific node. Therefore, in general, inpainting identifies a fixed number of patches as possible labels per node up to some fixed limit. Given the use of the 3D reconstruction and resulting geometric proxy to unify the multiple views of the images, there is no longer the requirement to search for an optimal patch over the complete image. Therefore, only labels from patches located within the initial patch size (400 pixels for an HD image) are examined across the witness cameras. This use of judicious label pruning allows the examination of only a small number of correlated patches across all the witness views, which improves efficiency with no reduction in accuracy.

4 Results and Discussion

We evaluate our inpainting algorithm over five public MVV sequences (subsec. 4.1) [1]. A set of foreground object mattes Ω are obtained for all viewpoints using [14] and hand labelling. Visual examples of source footage, mattes and inpainted regions produced by our algorithm are presented in Fig 6 and in the accompanying video (The video is available at `https://bit.ly/2DQYCw3`).

We quantify the accuracy of the sequences *Odzemok*, *Steps* and *Courtyard* via PSNR comparison with a static clean plate unavailable to the algorithm. Together with qualitative results for three sequences with a moving principal camera; *TuROM*, *Courtyard* and *Amphitheatre*, where it is infeasible to capture a clean plate for each frame. We also analyse the patch size and the number of witness views and the runtime performance against other approaches.

4.1 Multi-view Video Datasets

The sequence *Odzemok* was captured in an indoor studio using 9 HD cameras in a 180-degree arc, the background consists of multiple depths, with large swathes of the background occluded in the principal view over all time, including the sofa, wall and floor. *Steps* consists of 8 HD cameras in an outdoor setting in a 150-degree arc with multiple moving people, and a dynamic background of trees. The multiplanar geometry of the steps presents a challenging scenario for patch re-projection in the presence of a noisy geometric proxy. The sequence *Amphitheatre* uses the same scene and cameras as *Steps*, however with a dynamic mid-ground consisting of multiple moving and occluded people, ensuring a unique inpainting result is required for each frame. The sequence *Courtyard* is filmed in a small outdoor courtyard, with 14 cameras in a 180-degree arc and a moving hero camera and a large number of small objects in the scene. We show results using both a static and moving hero camera. The sequence *TuROM* consists of 6 static witness cameras in a 150-degree arc and a single moving hero camera in an outdoor setting with a single person performing a Range of Motion (ROM) sequence. This sequence features a white box behind the actor never seen from the hero camera. In terms of parameters, we use $M_{ph} = 400$, the initial patch size and the patch lifetime threshold $\tau = 5$.

4.2 Evaluation of Accuracy

We perform comparative evaluation of our method (**DoP-MRF-OF**) with a large number of baseline approaches (using their released code, but modified to use all camera views): 1) **FVVR**, a free-viewpoint video rendering technique [20]; 2) **Efros** and Freeman's patch-based inpaint-

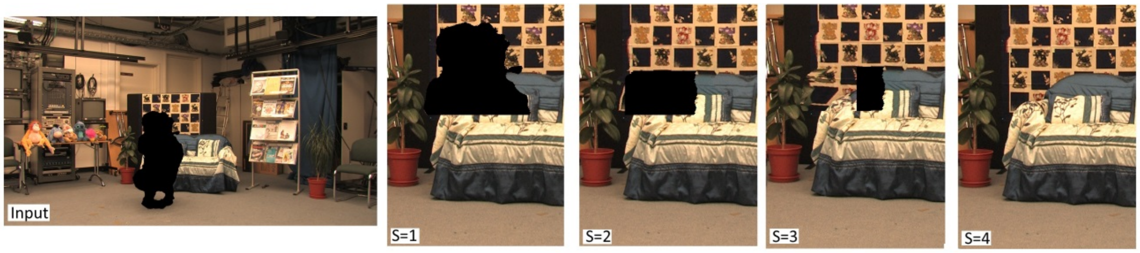


Fig. 5: The progressive refinement of inpainted texture as the multi-resolution algorithm increases scale $s = [1, 4]$, enabling large Ω_h to be filled.

ing [9] using patches sampled directly from all views; 3) **Single plane**, as per (2) with a single homography to reproject all texture patches to the principal view; 4) **Geometric proxy**, as per (3) but with full multi-planar geometric proxy reprojection; 5) **PatchMatch** [3] a monocular inpainting technique adapted to draw upon all camera views; 6) **Image Melding** [8], a monocular inpainting technique that incorporates geometric patch transformations from all camera views; 7) **PatchMatch3D** [27] the state of the art video inpainting using static and moving camera implementations. 8) **DoP-MRF**, a modification of (4) with a multi-resolution MRF and DoP to enforce coherent and plausible patch proposals; (9) **DoP-MRF-OF**, as per (8) but using our coarse to fine optical flow patch alignment. Fig. 6 presents representative visuals of our proposed method (9) against baseline approaches (2,5,6,7) and a ground truth clean plate for the static cameras, or an approximate camera view for moving cameras, indicated by a white star, additional discussion of the sequences is in Section 4.5 and a video of the sequences is available at `http://bit.ly/2pJEsun`). Mean average SSIM [36] of the inpainting result against the cleanplate is reported in Tbl. 1 and plotted in Fig. 7 over time for the *Odezmok* sequence, consistently the SSIM is higher for our proposed method.

4.3 Evaluation of Temporal Consistency

To evaluate the temporal consistency of the approach, we adopt the stability metric of Hua *et al.* [7] used in their work to assess flicker in video stylization. The resultant inpainted frame is divided into dense equi-sized patches. Each patch is compared against its spatial neighbours in the adjacent frames, and the maximum SSIM across the neighbourhood computed. That aggregated scores across all patches provides a measure of temporal coherence (inversely correlated to flicker) in the video. The temporal coherence of the approach and other state of the art methods is shown in Tbl. 2 and per frame over the *TuROM* and *Courtyard* (moving camera) sequences in Fig 8. Fig 9 visualises the false colour heat map for frames throughout the two sequences. The high SSIM for both our proposed approach, indicates that there is little change between the frames - therefore minimal artefacts or ”popping” of patches occurring through the temporal sequence. Fig 9 visualises the stability score over regions of the representative frames. Both PM3D and ImgMeld appear less coherent, indicated by the greater quantity of red pixels.

4.4 Perceptual User Study

In order to fully analyse the qualitative performance of our proposed approach, we performed a two user studies, analysing the performance of the inpainting against individual frames and short 40 frame sequence clips. The two modes, image and video were independent experiments each shown to 300 different users. For each test the user was shown the same sequence or frame with 3 inpainting techniques side by side, our proposed DoP-MRF-OF, Patchmatch3D [27] and image melding [8] (in a randomised order). The users were asked to ”select the highest quality video/image”, Tbl 3, shows the percentages of approaches chosen split over sequences It can be seen that our approach perform excellently, in particular in the video clips for the moving camera sequences TuROM and CourtyardMoving. For individual frames, TuROM is lower as the other approaches are still able to produce a pleasing (but blurred) result as shown in Figure 6. However, these results don’t represent the groundtruth scene as they ignore the white box.

4.5 Failure Cases

The only *failure* in *Odzemok* is at the top left of the sofa, where the geometric proxy is a poor approximation of scene structure. The alignment of the steps in *Steps* requires the coarse to fine optical flow alignment to provide near photo realistic in painting. The dynamic non-rigid background in *Amphitheatre* of trees and moving people demonstrates the impressive ability of our proposed approach to correctly inpaint the non-rigid moving objects. This is possible due to the updating of the geometric proxy on a frame wise basis, in conjunction with the patch freshness favouring recent patches from the witness views. Allowing our approach to work with further moving backgrounds such as vehicles or a beach, however smaller non rigid objects such as humans

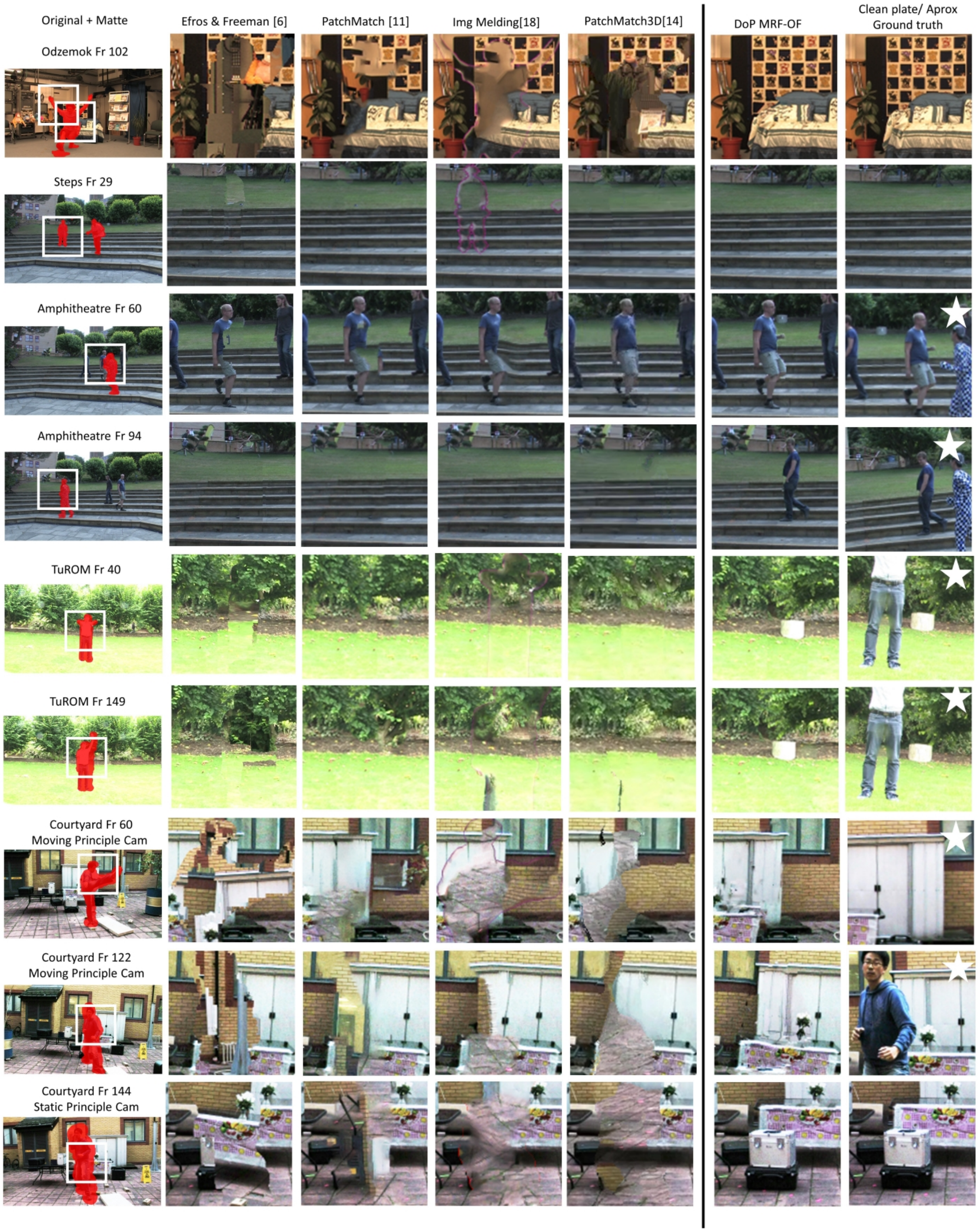


Fig. 6: Qualitative results of the proposed method **DoP-MRF-OF**, against other baselines (subsec. 4.2), including a ground truth clean plate or approximate moving camera view (indicated by a white star), discussion of the sequences is shown in section 4.5

Table 1: Accuracy quantified as SSIM against the groundtruth cleanplate (higher is better)

Seq	FVVR		Efros		1 plane		PtchMtch[11]		Geo Prox		ImgMeld[18]		PM3D[14]		DoP-MRF-OF	
	SSIM	σ	SSIM	σ	SSIM	σ	SSIM	σ	SSIM	σ	SSIM	σ	SSIM	σ	SSIM	σ
Odzemok	0.92	0.02	0.89	0.01	0.90	0.00	0.94	0.00	0.91	0.01	0.94	0.00	0.92	0.00	**0.98**	0.00
Steps	0.99	00.1	0.92	0.01	0.91	0.01	0.98	0.00	0.91	0.02	0.95	0.00	0.97	0.00	**0.99**	0.00
Curtyrd(static)	0.68	0.02	0.82	0.02	0.79	0.02	0.83	0.02	0.80	0.02	0.85	0.01	0.86	0.02	**0.92**	0.01

Table 2: Temporal consistency of approach across all sequences, measured through patch similarity computed by SSIM in a local neighbourhood on the previous frame. (Higher is better)

Seq	FVVR		Efros		1 plane		PtchMtch[11]		Geo Prox		ImgMeld[18]		PM3D[14]		DoP-MRF-OF	
	SSIM	σ	SSIM	σ	SSIM	σ	SSIM	σ	Prev	σ	SSIM	σ	SSIM	σ	SSIM	σ
Odzemok	0.94	0.00	0.94	0.02	0.92	0.01	0.95	0.01	0.92	0.02	**0.99**	0.00	**0.99**	0.01	**0.99**	0.00
Steps	0.98	0.01	0.98	0.02	0.97	0.02	0.97	0.01	0.96	0.03	**0.99**	0.00	**0.99**	0.00	**0.99**	0.00
Ampitheatre	0.90	0.03	0.89	0.03	0.91	0.03	0.92	0.02	0.91	0.03	0.96	0.02	0.96	0.01	**0.98**	0.01
TuROM	0.81	0.04	0.75	0.04	0.83	0.03	0.83	0.03	0.83	0.02	0.86	0.01	0.87	0.01	**0.90**	0.01
Curtyrd(Static)	0.87	0.02	0.95	0.01	0.93	0.02	0.88	0.02	0.92	0.02	0.92	0.01	**0.99**	0.01	**0.99**	0.00
Curtyrd(Move)	0.92	0.03	0.78	0.03	0.82	0.02	0.93	0.02	0.92	0.02	0.94	0.01	0.97	0.01	**0.98**	0.01

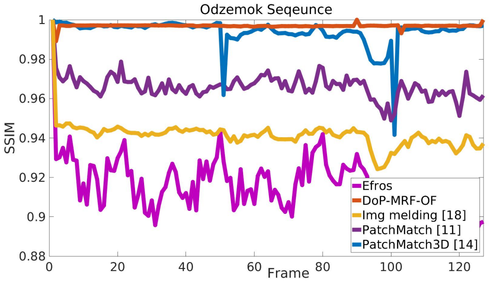


Fig. 7: Accuracy vs. time: per-frame SSIM against cleanplate for the *Odezmok* sequence.

can be harder to model geometrically, due to the resolution of the geometry incorrectly reprojecting a body part. The geometric proxy is updated on a frame by frame basis for the moving backgrounds and this can cause popping in the result, due to large scale geometric changes in the proxy. Also given the approach driven purely by patches, if none of the cameras can see an area due to occlusion there is likely to be an incorrect optimisation found.

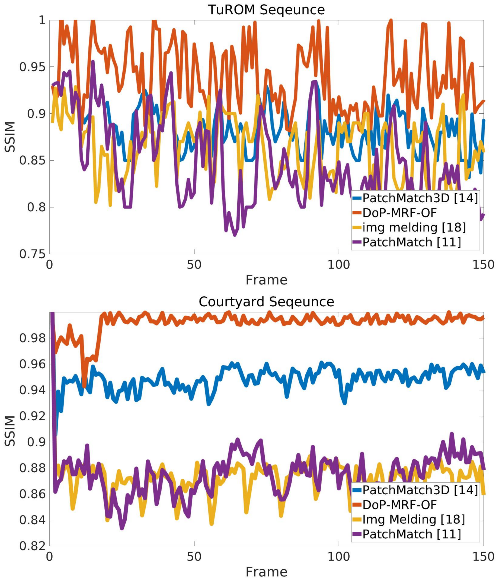


Fig. 8: Temporal consistency of *TuROM* and *Courtyard* sequences.

Our approach, **DoP-MRF-OF** provides both quantitatively and qualitatively clearer inpainting of the matte in comparison to the other approaches. The current state of the art temporal based video inpainting optimisation method, **PatchMatch3D** [27], is successful for the sequence *Steps*, with its repetitive structures like many other inpainting sequences, such as the beach umbrella. However while it attempts to minimise the visual spatiotemporal errors by smoothing the inpainted regions, compared to the groundtruth it struggles in preserving the actual structure of the background in *Odzemok* (the sofa), *Courtyard*(the lower part of the image) and *TuROM*(the white box and a temporal smoothness). The use of **PatchMatch** improves the visual appearance in comparison to the older approach by **Efros** and Freeman, however, similar to **PatchMatch3D**, a lack of geometry and depth constraints between views causes significant failures in the *Odzemok*, *Amphitheatre* and *TuROM* sequences. Use of a full geometric proxy (**DoP-MRF** improves the approach with the temporal and spatial

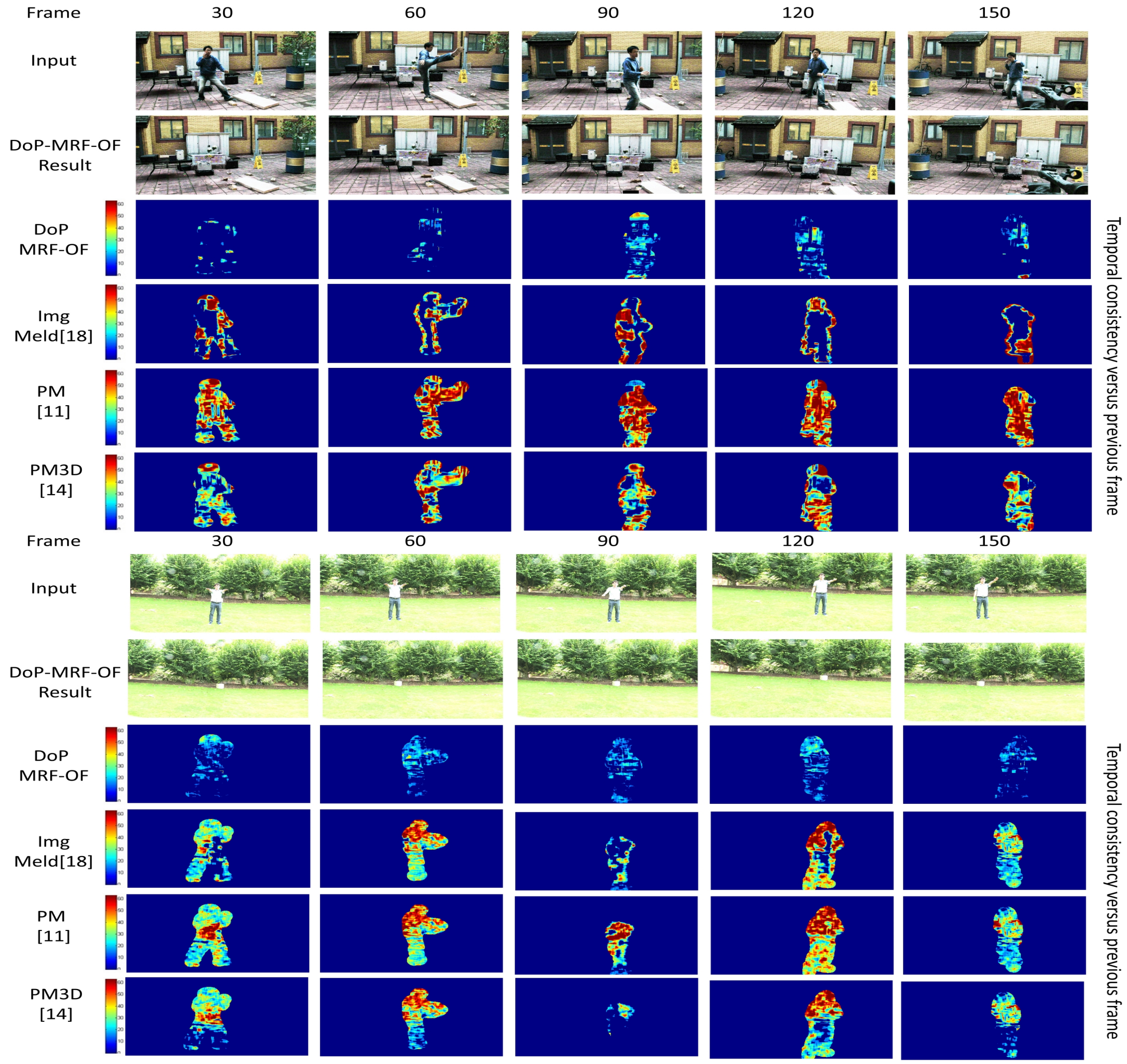


Fig. 9: Visualisation of the temporal instability of inpainted result using the stability score method of Hua *et al.* (red indicates regions of poor temporal stability)

constraints that the MRF framework provides, however, the alignment of patches is sometimes missed and in the case of *TuROM* the right background is never identified. This highlights the value of an MVV inpainting method; monocular approaches can only hallucinate missing detail when the actual environment is never observed (e.g. in *Odzemok*, *Amphitheatre* and *TuROM*). Analysis between **PatchMatch**, **PatchMatch3D**, **Image Melding**, **single plane** and **DoP-MRF-OF**, is shown in Fig. 10. The error of the inpainting patches against the background plate shown for *Odzemok* and *Courtyard*. It shows the minimal error of our approach against the clean plate, compared to the other image and video techniques, that especially struggle to inpaint the sofa and surprisingly in the case of **PatchMatch3D** the cloth.

4.6 The use of multi-view images

To demonstrate the importance of the multi-view aspect of the approach, we perform inpainting using a decreasing number of camera views on the *Odzemok* sequence, and the accuracy measured through the mean, min, max and

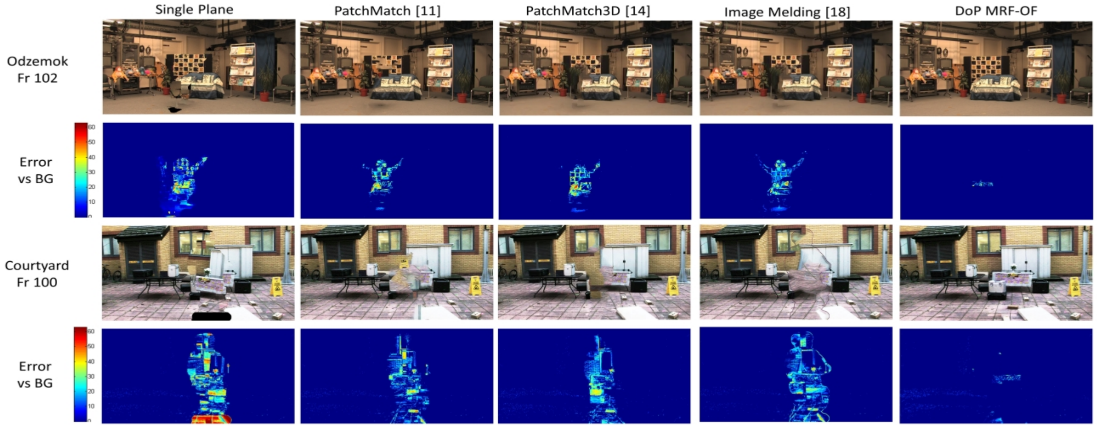


Fig. 10: Visualisation of the error vs captured clean plate, and patch cameras used for **DoF-MRF-OF**, **PatchMatch** and **Single Plane** and **PM3D**

Table 3: Qualitative inpainting User study, across all frames and video clips of the sequences, 300 users were asked to pick their ”highest quality” video/image, after inpainting with the 3 approaches

Seq	PM3D	Img Meld	DoP-MRF-OF
Single Frame Comparison			
Odzemok	2.1%	2.7%	**95.2%**
Amphitheatre	18.2%	10.4%	**71.4%**
TuROM	25.0%	15.0%	**59.4%**
Courtyard moving	2.1%	3.4%	**95.0%**
Courtyard static	1.5%	1.5%	**97.0%**
Video clip Comparison			
Odzemok	1.4%	2.4%	**96.2%**
Amphitheatre	22.9%	8.0%	**69.1%**
TuROM	9.7%	5.1%	**85.2%**
Courtyard moving	0.7%	3.2 %	**96.1%**
Courtyard static	4.8%	2.9%	**92.3%**

standard deviation of the PSNR for all view permutations. The results on the performance of the inpainting against the number of camera views are displayed in Fig 11.

It can be seen that excluding the hero camera view only (excl PC), makes no difference to the performance, however, once three or more cameras are removed the performance drops. Fig 12 confirms the importance of the additional views, by showing the average number of patches drawn from each camera view for the five sequences. It can be seen that a number of cameras are used, this is due to the geometric proxy warping the patches with respect to the scene, thus allowing them to be used effectively in the inpainting. Although the steps sequence does use a large number of patches from the hero camera (4), The steps sequence

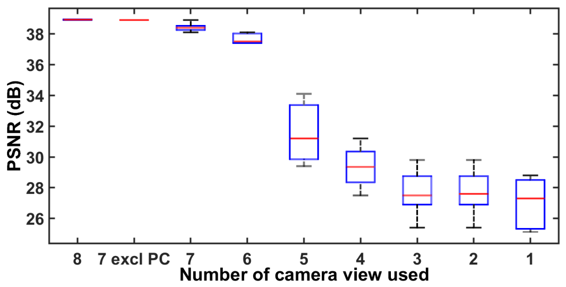


Fig. 11: Accuracy vs number of camera views used

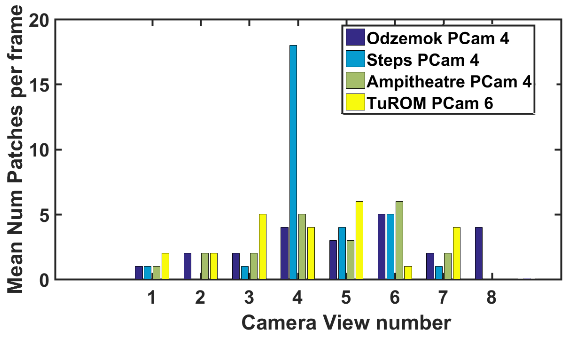


Fig. 12: Illustration of the mean of patches drawn from each camera view for the 4 sequences

has a large amount of similar low-frequency texture in the steps and background grass on the hero camera (in this case camera 4). Also, the subject to inpaint is quite far away meaning there are more patches in the hero camera, therefore it uses the hero camera to source many of the patches. Although excluding the hero camera, makes no difference on average to the performance of the approach as shown in Figure 4.8. The other sequences use the hero camera less as their sequences generally contain more occlusion of the hero camera due to the subject being larger or using moving cameras, these mean the witness views are needed more.

4.7 Dictionary of Patches

The use of a DoP is also key to the approach, Fig 13, shows the performance of the *Odzemok* and *Steps* sequences when the dictionary of patches lifetime threshold τ is varied. It can be seen that as the number of frames kept in the dictionary increases so does the performance until it stabilises at around five frames. Allowing patches from previous frames to be used but without the naive approach of testing all possible patches in the sequence, with its exponential increase in storage and computation costs. It is possible to remove the dictionary and optimise overall frames in a sequence, and the MRF has access to all possible patches, this can remove the flicker at the beginning of the sequence, indicated by the instability in the initial frames. However, the mean PSNR over the Odzemok and Courtyard sequences are unaffected at 38.9dB and 39.8dB respectively, despite a huge increase in the computational runtime cost of the approach, increasing the time per frame from around 1s to over 10s on average.

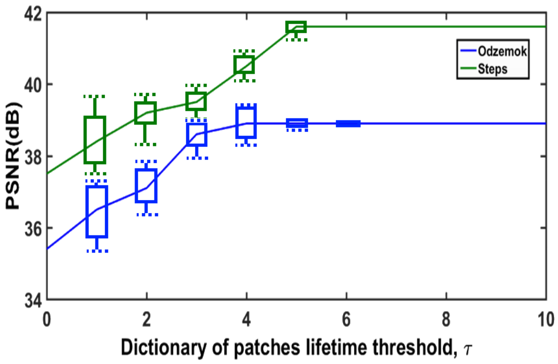


Fig. 13: Comparing performance for increasing the maximum frame lifetime of patches in the DoP τ

4.8 Patch size

The approach is based on image patches as these provide improved local structure, with minimal computational costs. It is possible to adjust the initial patch size, both lower and higher than the standard 400pixels for an HD video. Figure 14 shows how the PSNR changes for **Odzemok** and **Courtyard** for different initial patch sizes.

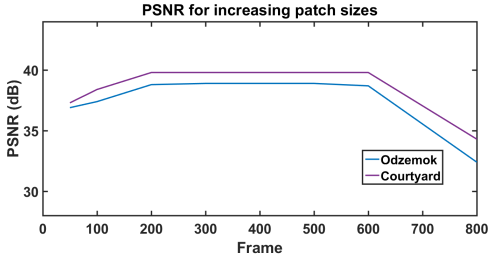


Fig. 14: The mean PSNR performance of Odzemok and Courtyard with a static hero camera with varying initial patch size

While Fig 15 qualitatively shows the change in the quality of inpainting for a region of frame 40 of TuROM as the initial patch size is changed

These examples show that as the patch size decreases dramatically, it starts to lose performance, as it loses the ability for a coherent globally optimised inpaint. Similarly, as it increases the patch size, there are less possible patches to use no corrupted by the matte area and the approach fails to inpaint at all.

4.9 Non-rigid Mid Ground Objects

When dynamic or static midground objects are present in the sequence, the use patch depth in the unary term of the MRF dramatically improves the inpainting quality (while un-affecting sequences without midground objects. Fig 16 shows the frames from the *Amphitheatre* sequence from Fig 6 when the depth is and isn't considered in the MRF. It can be seen that without depth the dynamic objects are also ignored, and the background is inpainted incorrectly, as shown in Fig 6 this occurs with other approaches such as **PatchMatch3D** too. However, with the depth term, the midground objects which cannot be seen from the hero camera are inpainted correctly.

4.10 Run-time Performance

Often within optimisation frameworks, the per frame runtime cost of the inpainting process is very high and not practical for realistic use especially for video, taking hours per frame. Tbl 4 shows the mean time per frame in seconds across the different approaches for two of our HD video sequences.

As can be seen compared to the approaches **PatchMatch3D**(PM3D) and **Image Melding**(imgMeld), our approach **DoP-MRF-OF** show a speed-up of at least an order of magnitude . This advantage is significant because not only is the algorithm more practical to use, given that a

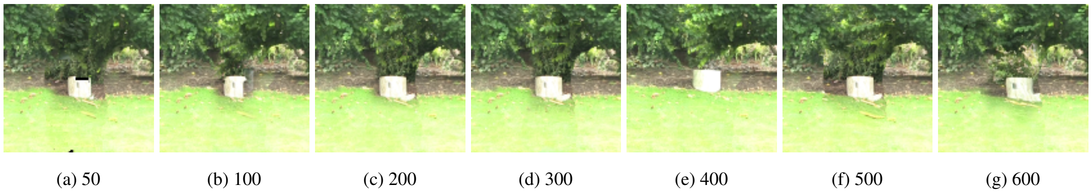

(a) 50 (b) 100 (c) 200 (d) 300 (e) 400 (f) 500 (g) 600

Fig. 15: Comparing performance as the initial patch size is changed for TuROM Fr 40. We use a 400-pixel size, as this provides a compromise between availability of patches and availability of local structure.

Seq	FVVR	Efros	1 plane	PM	Geo Prox	DoP-MRF	ImgMeld	PM3D	DoP-MRF-OF
Odzmk	0.5	0.8	0.8	0.8	0.8	0.8	4710	126	1.2
Curtyd	0.4	0.7	0.9	1.7	0.8	0.9	4920	131	1.5

Table 4: Comparing Mean time per frame (seconds) of our approach **DoP-MRF-OF** against previous works on the *Odzemok* and *Courtyard sequences*

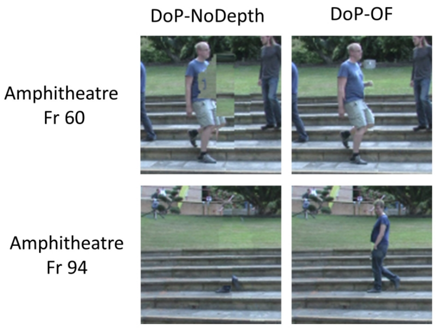


Fig. 16: Comparing the approach with and without optimisation from as defined in sec 3.2

video sequence will contain many frames, but it has state of the art performance compared to this other approaches. The Multi-resolution solution and selective candidate patches together with the carefully constrained dictionary label set in the MRF allows for a near real time operation with a state of the art performance.

5 Conclusion

We have presented the first algorithm for video inpainting over wide baseline MVV footage. Uniquely our MVV approach can inpaint regions of the principal view that are never visible in that view (e.g. the plinth in *TuROM* or people in *Amphitheatre*). Key technical contributions were an optimisation framework for disambiguating patch selection from an over-complete DoP sampled from principal and witness views. Texture patches from the latter were reprojected to the principal view using a coarse geometric proxy recovered from the scene. The DoP was constructed over time and patches culled based on age and popularity to retain tractable execution times. The MRF constrain patch choices to encourage visual plausibility, bias toward more recent patches, and innovative depth based spatiotemporal coherence. We evaluated using five public MVV sequences and results showed excellent qualitative and quantitative performance when compared to baseline monocular and multiple-view techniques. Future work will evaluate performance over backgrounds exhibiting more substantial motion, considering the possibility of a dynamically updating geometric proxy e.g via Kinect Fusion [26]. It would also be interesting to explore the effect of near-camera objects in the scene either directly behind or directly occluding the foreground object and how reasoning about visibility could be factored into the unary term of the MRF.

Acknowledgments

This research was supported by InnovateUK projects; REFRAME & TotalCapture, the BBC, the Imaginarium Studios London, and Imagineer Systems (BostonFX). we are grateful to Chris Russell for input on the MRF exposition.

References

[1] CVSSP3D: Multiple viewpoint video repository. `http://cvssp.org/cvssp3d`. Last Accessed: 2015-11-05.

[2] Photoshop content aware tool. In *http://www.photoshopessentials.com/photo-editing/content-aware-fill-cs5/.*, 2016.

[3] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing,. In *Proc. ACM SIGGRAPH*, 2009.

[4] C. Barnes, F.-L. Zhang, L. Lou, X. Wu, and S.-M. Hu. Patchtable: Efficient patch queries for large datasets and applications. In *ACM Transactions on Graphics (Proc. SIGGRAPH)*, Aug. 2015.

[5] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image in-painting. In *Proc. ACM SIGGRAPH*, 2000.

[6] J. Besag. On the statistical analysis of dirty pictures. *Royal Statistical Society B*, 48:259–302, 1986.

[7] D. Chen, J. Liao, L. Yuan, N. Yu, and G. Hua. Coherent online video style transfer. *arXiv preprint arXiv:1703.09211*, 2017.

[8] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen. Image melding: combining inconsistent images using patch-based synthesis. In *ACM TOG*, 2012.

[9] A. Efros and W. Freeman. Image quilting for texture synthesis and transfer. In *Proc. SIGGRAPH*, 2001.

[10] A. Efros and T. Leung. Texture Synthesis by non-parametric sampling. In *Proc. Intl. Conf. on Computer Vision (ICCV)*, 1999.

[11] Y. Eshet, S. Korman, E. Ofek, and S. Avidan. Dcsh-matching patches in rgbd images. In *ICCV*, 2013.

[12] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In *Proc. ICCV*, 2009.

[13] M. Granados, K. Kim, J. Tompkin, O. Grau, J. Kautz, and C. Theobalt. How not to be seen: Object removal from videos of crowded scenes. *Proc. Eurographics*, 2012.

[14] J.-Y. Guillemaut and A. Hilton. Joint multi-layer segmentation and reconstruction for free-viewpoint video applications. *IJCV*, 1(93):73–100, 2011.

[15] J. Hays and A. Efros. Scene completion using millions of photographs. In *Proc. ACM SIGGRAPH*, 2007.

[16] A. Hervieu, N. Papadakis, A. Bugeau, P. Gargallo, and V. Caselles. Stereoscopic image inpainting: Distinct depth mapsand images inpainting. In *ICPR*, 2010.

[17] S.-M. Hu, F.-L. Zhang, M. Wang, R. R. Martin, and J. Wang. Patchnet: A patch-based image representation for interactive library-driven image editing. *ACM Transactions on Graphics*, 2013.

[18] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf. Image completion using planar structure guidance. In *ACM Transactions on Graphics*, volume 33, page 129. ACM, 2014.

[19] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf. Temporally coherent completion of dynamic video. *ACM Trans. Graph.*, 2016.

[20] H. E. Imre, J.-Y. Guillemaut, and A. Hilton. Moving camera registration for multiple camera setups in dynamic scenes. In *Proc. BMVC*, 2010.

[21] N. K. Kalantari, E. Shechtman, C. Barnes, S. Darabi, D. B. Goldman, and P. Sen. Patch-based High Dynamic Range Video. *SIGGRAPH Asia 2013*, 32(6), 2013.

[22] S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High dynamic range video. *ACM Transactions on Graphics*, 22(3):319–325, 2003.

[23] F. Klose, O. Wang, J.-C. Bazin, M. Magnor, and A. Sorkine-Hornung. Sampling based scene-space video processing. In *ACM Transactions on Graphics*, volume 34, page 67. ACM, 2015.

[24] Y. Liu and V. Caselles. Exemplar-based image inpainting using multiscale graph cuts. *IEEE Trans. on Image Processing*, pages 1699–1711, 2013.

[25] B. Morse, J. Howard, S. Cohen, and B. Price. Patchmatch-based content completion of stereo image pairs. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, pages 555–562, 2012.

[26] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proc. IEEE ISMAR*, 2011.

[27] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. Perez. Video In-Painting of Complex Scenes. *SIAM Journal of Imaging Science*, 7:1993–2019, 2014.

[28] K. Patwardhan, G. Sapiro, and M. Bertalmio. Video inpainting under constrained camera motion. *IEEE Trans. on Image Processing*, 2(16):545–553, 2007.

[29] T. Pouli, E. Reinhard, and D. W. Cunningham. *Image Statistics in Visual Computing*. A. K. Peters, Ltd., Natick, MA, USA, 1st edition, 2013.

[30] Y. Pritch, E. Kav-Venaki, and S. Peleg. Shift-map image editing. In *ICCV'09*, 2009.

[31] T. Thonat, E. Shechtman, S. Paris, and G. Drettakis. Multi-view inpainting for image-based scene editing and rendering. In *3DV'16*, pages 351–359. IEEE, 2016.

[32] C. Vazquez, W. J. Tam, and F. Speranza. Stereoscopic imaging: filling disoccluded areas in depth image-based rendering. In *Proc. SPIE*, 2006.

[33] M. V. Venkatesh, S. S. Cheung, and J. Zhao. Efficient object-based video in-painting. *Pattern Recognition Letters*, 2(30):168–179, 2009.

[34] L. Wang, H. Jin, R. Yang, and M. Gong. Stereoscopic inpainting: Joint color and depth completion from stereo images. In *CVPR'08*, 2008.

[35] M. Wang, X.-J. Zhang, J.-B. Liang, S.-H. Zhang, and R. R. Martin. Comfort-driven disparity adjustment for stereoscopic video. *Computational Visual Media*, 2016.

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004.

[37] Y. Wexler, E. Shechtman, and M. Irani. Space-time image completion. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2004.

[38] F.-L. Zhang, J. Wang, E. Shechtman, Z.-Y. Zhou, J.-X. Shi, and S.-M. Hu. Plenopatch: Patch-based plenoptic image manipulation. *IEEE TVCG*, 2017.

[39] Z. Zhu, H. Huang, Z. Tan, K. Xu, and S. Hu. Faithful completion of images of scenic landmarks using internet images. *IEEE TVCG*, 2016.

[40] Z. Zhu, R. R. Martin, R. Pepperell, and A. Burleigh. 3d modeling and motion parallax for improved videoconferencing. *Computational Visual Media*, 2016.

Andrew Gilbert is a Senior Lecturer at the University of Surrey and his current research interests include human pose estimation. inpainting, real-time visual tracking, human activity recognition, and Intelligent Surveillance. He is a member of the BMVA Executive committee and coordinates the national BMVA technical meetings. He is the author of over 40 papers, journals, and book chapters.

Mathew Trumble is a PhD student in CVSSP at the University of Surrey. He completed a bachelor degree in Computer Science at the University of Surrey in 2014. His research interests include deep learning, multiview reconstruction and human pose estimation.

Adrian Hilton BSc(hons),DPhil,CEng, is a Professor and Director of CVSSP at the University of Surrey. His interest is in robust computer vision to model and understand real world scenes and his work in bridging-the-gap between real and computer generated imagery combines the fields of computer vision, graphics and animation. He leads several EU and UK/ industry projects and holds a 5-year Royal Society Wolfson Research Merit Award.

John Collomosse is a Professor within CVSSP at the University of Surrey and a Visiting Prof. at Adobe Systems' Creative Intelligence Lab. John's research fuses Computer Vision, Graphics and Machine Learning to tackle Big Data problems in visual media, with emphasis on post-production in the creative industries and intelligent algorithms for searching and rendering large visual media repositories.