Optimal Representation of Multi-View Video

Marco Volino
m.volino@surrey.ac.uk
Dan Casas
d.casas@surrey.ac.uk
John Collomosse
j.collomosse@surrey.ac.uk
Adrian Hilton
a.hilton@surrey.ac.uk

Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, UK

Introduction: Multi-view video acquisition is widely used for reconstruction and free-viewpoint rendering (FVR) of dynamic scenes. Current approaches to FVR resample directly from the captured multi-view images at each time frame, achieving a high level of photo-realism but requiring storage and transmission of multi-video sequences. This is prohibitively expensive in both storage and bandwidth required for multiple video streams limiting applications to local rendering on high-performance hardware. This paper addresses the problem of optimally resampling and representing multi-view video to obtain a compact representation without loss of the view-dependent dynamic surface appearance.

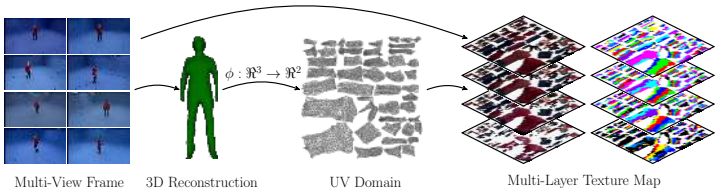


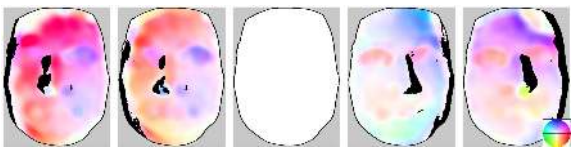
Figure 1: Overview of the resampling of multi-view video to a multi-layer texture video

Representation and Optimisation: Fig. 1 shows an overview of the proposed approach taking as input a set of camera images and an aligned mesh sequence. Texture coordinates, a 3D-2D mapping, are defined and the multi-view images are resampled into a hierarchy of texture maps with the views of each facet ordered by visibility. Optimal resampling from multiple views requires spatial and temporal coherence of the representation. The problem can be cast as a labelling problem where we seek the mapping $L : F \rightarrow C$ from the set of mesh facets F to the set of cameras $C = \{1...N_C\}$ which assigns a camera label $l_f \in C$ to each facet $f \in F$. We formulate the computation of the optimal labelling $L(t)$ as an energy minimisation of cost:

$$E(L(t)) = \sum_{\forall t} (E_v(L(t)) + \lambda_s E_s(L(t)) + \lambda_t E_t(L(t), L(t+1)). \quad (1)$$

where $E_v(L(t))$ is the unary visibility cost for all faces F to be assigned camera labels $L(t)$ at time t, $E_s()$ is the spatial coherence cost which enforces consistent camera labelling between adjacent mesh facets, $E_t()$ is the temporal coherence cost which enforces temporal coherence of the camera labelling, finally λ_s and λ_t are weighting terms for the spatial and temporal smoothness functions.

(a) Textured geometry C_i $i = 1...5$ (b) Rendered Image I_3^j for $j = 1...5$)



(c) Optical flow $O_{i \rightarrow j}$ between rendered image I_3 and I_3^j for $j = 1..5$. Optical flow colour mapped to direction (bottom right, maximum flow 5 pixels)

Figure 2: Results of surface-based optical flow alignment of appearance from multiple views

Multi-View Alignment: Simple projection and blending of camera views using the approximate reconstructed mesh geometry leads to blurring and ghosting artefacts. These artefacts are caused by misalignment between overlapping camera images projected onto to mesh surface from inaccurate geometry and camera calibration. In order to minimise these artefacts, we use optical flow based image warping to correct misalignments before sampling into the texture domain. To establish optical flow between camera views, we first render the geometry from the viewpoint of camera C_i and projectively texture the geometry using the image of camera C_j for all N_C cameras. This results in N_C^2 rendered images, R_i^j, which denotes the image rendered from the i^{th} camera viewpoint using the j^{th} camera image, Fig. 2(a) and (b). An optical flow correspondence field, $O_{i \rightarrow j}$, is computed between the rendered image $R_i = R_i^i$ and R_i^j where $i \neq j$. A binary confidence score is assigned to each flow vector, black indicates areas where occlusion or depth discontinuities occur these are assigned a zero confidence scores, Fig. 2(c). The magnitude of the correction vector is given by the weighted average of all visible and high-confidence flow vectors on the surface.

Results: Optimal resampling of the captured multi-view images as a layered texture map representation is achieved by combining the optical flow alignment of the captured images on the reconstructed surface with the spatio-temporal optimisation of camera label assignments for each mesh facet. Fig. 3 shows two examples of the multi-view alignment: (a) a texture map layer from dataset Dan. (b) First three layers from Cloth dataset blended together. This demonstrates that the approach corrects misalignment which reduces ghosting and blur artefacts during rendering. The representation is evaluated in terms of rendering quality and required storage when varying the size of the texture map and number used. We show that only 3 texture layers are required to maintain view dependence during rendering and no significant increase in quality occurs when using a texture size above 1024. This results in a >90% reduction in the required storage when compared to the captured data.

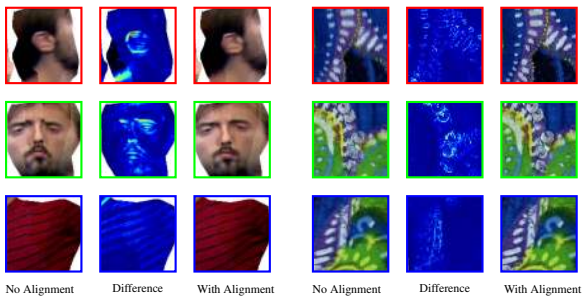


Figure 3: Results of multi-view optical flow based alignment

Conclusion: A method is presented for optimisation of the resampling from multi-view video sequences of a reconstructed surface into a multi-layer 2D texture map representation to obtain a compact, spatially and temporal coherent representation that minimises the loss of information from the captured data to maintain FVR quality. Spatio-temporal optimisation is combined with a surface-based optical flow alignment to significantly reduce the storage footprint and minimise artefacts due to errors in geometry and camera calibration. This demonstrates that the proposed approach results in an efficient representation that preserves the visual quality of the captured multiple view video for FVR whilst achieving approximately >90% reduction in size.