https://doi.org/10.2352/ISSN.2470-1173.2017.15.DPMI-064

Panoramic Background Estimation from RGB-D Videos

Christos Bampis[1]***, Gowri Somanath***[2]***, Oscar Nestares***[2]***, Jiajie Yao***[3]***;***
[1] ***University of Texas at Austin, USA,*** [2] ***Computational Imaging Lab, Intel Labs, USA.*** [3] ***Intel Corporation, China.***

Abstract

We propose a novel hybrid framework for estimating a clean panoramic background from consumer RGB-D cameras. The method explicitly handles moving objects, eliminates distortions observed in traditional 2D stitching methods and adaptively handles errors in input depth maps to avoid errors common in 3D based schemes. It produces a panoramic output which integrates parts of the scene as captured from the different poses of the moving camera and removes moving objects by replacing them with their correct background information in color and depth. A fused and cleaned RGB-D has multiple applications such as virtual reality, video compositing and creative video editing. Existing image stitching methods rely on either color or depth information and thus suffer from perspective distortions or low RGB fidelity. A detailed comparison between traditional and state-of-the-art methods and the proposed framework demonstrates the advantages of fusing 2D and 3D information for panoramic background estimation.

1. Introduction

Consumer and prosumer photography has seen an increase in the last decade with a large share coming from mobile photography and social media. Capturing moments with family and friends for quick sharing on the web is now being interjected with the need to share creatively edited versions. For instance, cinemagraphs (sequences where the relative speeds of objects in the scene have been changed from the original input) and video compositing are some of the popular video editing effects, but today they require professional setup and manual effort. The availability of 3D and multi-camera systems such as Intel RealSense [2] and Google Tango [1] in mobile devices can help automate many of these tasks. Many applications benefit from having a clean RGB-D of the static scene background, for re-composition of objects or stereo view generation for VR applications. A clean and panoramic background image (see Fig. 1) is defined as one that retains the static background color and depth, and is composed of all parts of the scene as revealed by a moving (panning or rotating) camera. We develop a framework to estimate such a background from videos captured from handheld mobiles devices with such RGB-D sensors. An overview of the proposed framework is shown in Fig. 2.

2. Related work

Panoramic 2D image stitching has been extensively studied and is now common in most mobile devices. Typically, a special mode is activated to take multiple overlapping pictures or panning video of the scene, which are seamlessly stitched to form the composite panorama [5, 12]. Most 2D based stitching methods assume that the scene is sufficiently far from the camera or does not contain many depth layers i.e. parallax can be ignored. This assumption allows using affine or homography transforms for the composition. However, if the scene is not roughly planar then the homography assumption is violated and stitching artifacts like broken image structures or ghosting due to moving objects appear. A homography can also be applied if the camera undergoes a pure rotation about the center of projection, however in practice this is hard to achieve in hand-held captures. To alleviate these artifacts, seam cutting approaches [4, 7] try to find the best possible seam to stitch two images and use image blending [10, 6] to create visually appealing results. These methods are able to partially tackle the misalignments given the underlying homography assumption. Zaragoza *et al* [13] introduced the idea of location-dependent projective warps and Lin *et al* [8] used a smoothened affine transform. These spatially-varying warping algorithms were shown to handle parallax better than homography. However in [14] it was demonstrated that the previous methods cannot address cases with large parallax. Therefore, an improved local stitching method is proposed which combines content-preserving warping and seam cutting.

Recently, using depth information has become more popular due to the reduced cost of depth cameras [9]. Exploiting the depth information overcomes the homography restriction and reduces the perspective distortions of 2D methods. Simultaneous Localization and Mapping techniques are used for real-time camera pose estimation, tracking and dense reconstruction of scenes. Salas-Moreno *et al* [11] observed that many scenes consist of specific objects/structures and developed a 3D object based scene representation. Newcombe *et al* [9] developed the first real-time, dense volumetric scene reconstruction using a hand-held Kinect depth sensor. Different from these 3D reconstruction and fusion works which target high quality 3D scans of a scene, our work focuses on stitching regular user videos where each part of the scene may be only seen for a few frames.

More related to our work, stereoscopic stitching [15] was applied on the stereo disparity along with the two pairs of stereo images. However, this method does not handle moving objects which results in artifacts. As shown in Figs. 3 and 9, previous works have errors when there are moving objects, large parallax, or in the presence of common errors or missing depth data. We propose a hybrid approach which deals with those problems and produces visually pleasing outputs required for the previously described creative media effects.

3. Overview of the proposed method and the dataset

To the best of our knowledge this is the first method that combines many sources of information: color, depth and object segmentations to generate a clean panoramic color and depth image from commodity 3D camera videos. An overview of the proposed method is shown in Fig. 2. Given a new RGB-D frame, the

Figure 1. *Overview of the proposed method which uses RGB-D and object segmentations to output a clean RGB-D panoramic scene. The red box denotes the first frame and the green outline shows the moving object in the reference frame.*

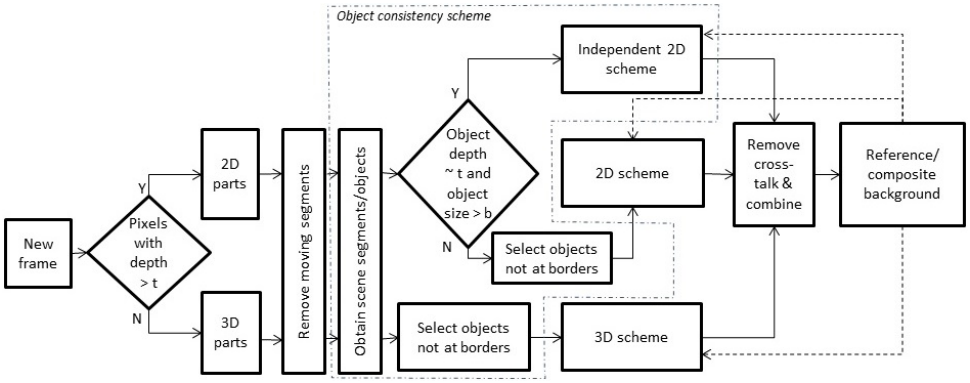

Figure 2. *Overview of the proposed framework.*

Figure 3. *Examples showing failure cases of traditional methods. From top to bottom: perspective distortion in the sofa on the right side (2D stitching), ghosting artifact due to a moving person on the left side (2D stitching) and low RGB fidelity on the right side (3D stitching).*

proposed method divides its pixels into two subsets: pixels that use a 2D transform pipeline and those that use a 3D transform. Both transforms are estimated using keypoint matching between a new and the reference RGB-D frame which is being updated over time. Then, we identify moving objects of a scene via RGB-D segmentation [16] and remove them from these disjoint pixel sets. This is necessary to eliminate motion related ghosting artifacts and perform background cleaning and filling. Next we apply scene segmentation to divide the frame into smaller components which are referred to as objects. These scene segments are used to process the two pixel subsets (assigned to 2D, 3D transformations respectively). Pixels that belong to an object that reaches the image borders are not immediately processed. The pixels belonging to an object are transformed together in one iteration to preserve object structure. We discuss how we handle special cases of this condition in further sections. An additional condition is considered for the 2D subset: objects that are relatively large and at a depth close to the sensor range limit are separated from the rest of the 2D subset and follow an independent 2D transform. All three cases (2D, 3D transformed and independent 2D transformed pixels) are then combined into a single composite frame. This frame becomes the reference frame for the next iteration. We provide details for each module in further sections.

To evaluate the proposed method we created a comprehensive dataset of video sequences captured by a hand-held Intel RealSense Snapshot device (Dell Venue 8) covering various natural scenarios. We included videos with different camera motions (natural panning, rotation, etc.), large viewpoint changes, combination of stationary and moving objects at various depths within and beyond the depth range, and variations of object textures and activities. Examples of our dataset can be seen in Fig. 4 (all figures are best viewed in color on a monitor). Darker regions in the disparity maps indicate the regions where depth information was missing or was measured with very low confidence. Given that an end user application is targeted, we compare the different methods through visual inspection for the problems discussed previously.

4. Proposed method

Fusing 2D and 3D information

As discussed previously using 2D or 3D information alone leads to errors and visual distortions (Fig 3). An adaptive fusion scheme can effectively combine the advantages of both approaches: the 3D information is exploited only when it is of good quality; otherwise a 2D scheme is preferable. For every input RGB-D frame, a simple decision for every pixel is made: if this pixel is within the sensor range and has a reliable depth

Figure 4. *Some frames and disparity maps from our typical input sequences. Top row: 'Jump' sequence (handheld panning camera following a fast moving object), Middle row: 'Penguin2' sequence (static scene with challenging surfaces for depth, rotating camera), Last row: 'Stanford selfie' sequence (scene with moving objects within and outside camera depth range, handheld camera rotating nearly 360 degrees).*

Figure 5. *Hybrid 2D/3D approach (left to right): a frame from the 'Penguin1' sequence, disparity map, map indicating 2D and 3D parts in the hybrid scheme. Far objects and near objects with unreliable depth estimates (blue regions) are being fed to the 2D pipeline.*

estimate, it will be transformed using a 3D transform, else we use a 2D transform (see Fig. 5). Prior to transforming the pixels from each category (2/3D), moving scene segments are eliminated from the transformation estimations as described in the next section. For the 2D transformation a traditional feature matching and RANSAC approach is used to create the transformation ($\mathbf{R}_{2D}$, $\mathbf{t}_{2D}$) where $\mathbf{R}_{2D}$ is a 2×2 rotation matrix and $\mathbf{t}_{2D}$ a 2×1 translation vector. Likewise using the available depth information the 3D transformation ($\mathbf{R}_{3D}$, $\mathbf{t}_{3D}$) is created. The pixels are re-projected to the reference coordinate system using the above transformations. If multiple pixels are transformed to the same target location in the composite frame they are discarded under the assumption that this is caused by an unreliable input depth measurement. As a final step, small regions with missing RGB values due to the quantization during the warping are filled by interpolation. The same steps (except for the final interpolation) are applied to the disparity/depth map to get the fused depth composite. The converted depth/point-cloud of the current frame is projected onto the reference frame to generate the correct disparity in the reference coordinate frame for the disparity composite.

Background Cleaning

Since the transforms are calculated from stable features, moving objects are incorrectly transformed and result in the ghosting artifacts observed in previous methods. In our proposed method we create a clean background where moving parts of the scene have been removed. Therefore, moving scene segments are detected and excluded from any transformation to the RGB-D composite frames. Using RGB-D segmentation [16] moving objects are detected by first selecting an area of interest in the reference frame. Following frames are initialized by transferring the previous frame segmentation mask using optical flow. Let R and N be the reference and new frame's moving part respectively in the reference coordinate system U. Then, consider the

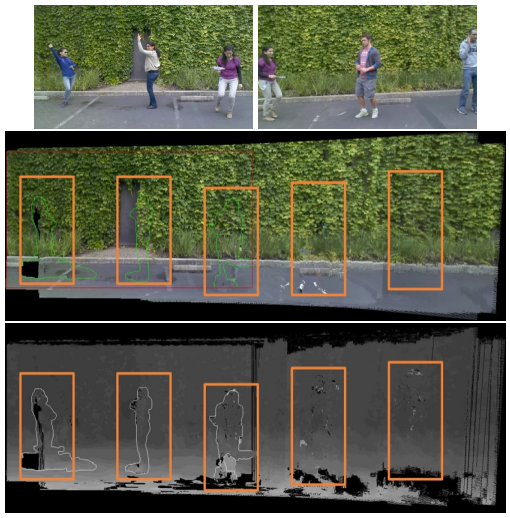

Figure 6. *Top row: Two sample frames from a panning video of five moving persons. Middle and bottom row: results from our RGB-D fusion system. Orange boxes indicate regions of moving objects correctly filled with background information. See Section 'Background Cleaning' for more details.*

area $F = R \cap N^c$ (where $(.)^c$ denotes the set complement) which can now be filled with a clean background due to the object motion. Through this iterative cleaning procedure, we can effectively remove moving segments.

As an example, Fig. 6 shows the cleaning process of five different moving users as viewed over the hand-held panning video sequence 'Multipeople'. The proposed module effectively removes all five moving objects. These regions are also excluded from transformation estimation and warping. The contribution of the background cleaning step is twofold: a clean background of the scene is created by filling in holes caused by moving objects and both the 2D and 3D based schemes are refined since the frames are now being matched only on features from the static background and hence model camera motion correctly. Further, the proposed framework is able to remove moving objects that appear not only in the original frame, but also as they are being introduced in the scene (see Fig. 6).

Object Consistency module

The modules discussed thus far handle scenes with multiple depths and moving objects. Another visual error observed is

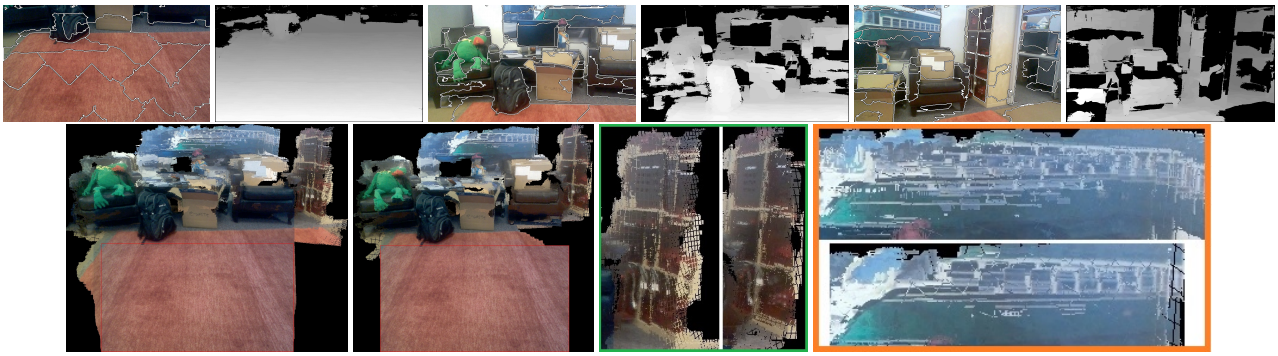

Figure 7. *Top row: RGB-D pairs and overlayed segmentation from the 'Penguin2' sequence. Bottom row: without (first) & with (second) object consistency. Distortions are greatly reduced on the top and right side of the scene as shown by the last two enlarged image pairs.*

distortion of objects, that can occur due to errors in input depth or inconsistent transformation of parts of the objects in multiple frames (see Fig. 7). This observation motivates another contribution: an 'object consistency' module whose goal is to maintain the object structures in a scene. By applying scene segmentation, coherent parts or 'objects' in the scene can be obtained. The term 'objects' loosely refers to segments obtained from the segmentation algorithm and not always to semantic objects. Therefore the segmentation specificity is not critical; only a set of relatively representative structures or objects inside the scene is needed. Two segmentation schemes are used to obtain such objects: multi-label RGB-D segmentation [16] and a RGB only segmentation using SLIC [3]. Their difference lies in that the former seeks to identify objects that are consistent both in color and depth whereas the latter depends only on their color properties. The proposed pipeline uses the 3D (RGB-D) segmentation for frames with reliable depth and uses SLIC instead for frames with depth measurements beyond the camera range. This decision is made independently on a frame by frame basis: if more than $p = 70\%$ of the frame has depth larger than the sensor range, then SLIC is used. The results are not sensitive to the value of p as SLIC usually yields satisfying results for our needs and the RGB-D scheme is used to further exploit the depth information of the scene.

In order to minimise distortions within a segment/object, pixels are selectively transformed based on the following criteria. Consider a particular pixel and the object it belongs to and examine if the object intersects with the image boundaries. If so, this pixel is not transformed until the whole object appears in the following frames due to the camera motion. If the object appears as a whole the transformation of this pixel is allowed. As shown in Fig. 7, the object consistency module keeps most of the scene structure intact such as the bus on the background poster, as well as the box and the storage cubes on the right.

Independent 2D transform

When using the object consistency module, some parts of the scene may not be transformed unless they appear as a whole object inside the frame boundaries (see the parts of the carpet in Fig. 7). This can be alleviated when more frames are considered while updating the reference frame. However, it may occur that very large background objects (such as the wall in Fig. 8) are never transformed since they are consistent enough (color and depth wise) to form a large scene segment that always touches the

Figure 8. *Top: using strict object consistency can lead to large missing sections of background (here right side parts of the wall and floor). Bottom: parts recovered by transformation through the use of the independent 2D module. See Section 'Independent 2D transform'.*

frame borders. To handle this, we define large segments as those with an area larger than 10% of the frame's size and with depth close to the sensor maximum range. We transform these regions independently of the rest of the scene i.e. find the $(\mathbf{R}_{2D}, \mathbf{t}_{2D})$ pair only for this particular region. Due to their depth and color, these segments are roughly planar and/or their depth is relatively large hence a 2D transformation suffices. As shown in Fig. 8, the independent transform scheme helps to bring in large parts of the wall on the background which would be otherwise excluded from the transformation. In Fig. 8, all the three contributions mentioned so far are included: background cleaning for moving objects, object consistency so that the scene's structure remains intact and the independent 2D transform to bring in large planar objects.

5. Experiments on RGB-D dataset

To demonstrate the advantage of the different modules in our proposed scheme, we compare our scheme with three other methods: state of the art parallax tolerant stitching method developed in [14, 15], 2D image stitching [5] and 3D image stitching based on RGB-D. Due to lack of space, only a few frames from each video are shown. First, consider the comparison between the parallax tolerant stereo stitching method and the proposed method in Fig. 9. The former was specifically built to handle moderate parallax in the scene, however it cannot handle the moving user and suffers both from a ghosting artifact and a visually unpleasant blending effect. By contrast, the background cleaning module in our scheme removes the moving user by filling in with correct

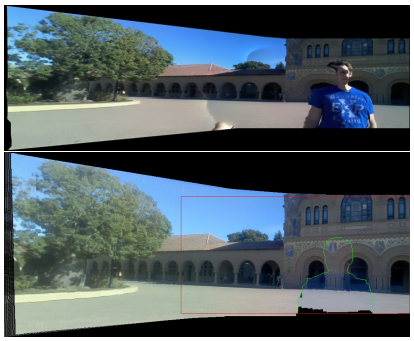

Figure 9. *Top: Distortions and ghosting artifacts in the state of the art parallax tolerant stitching method of [14, 15]. Bottom: clean output from our proposed pipeline. See Section 5.*

background information. The selected scene has the majority of the background beyond the depth camera range, hence both methods rely on 2D information resulting in some of the perspective distortion.

In Fig. 10 we compare our hybrid pipeline with the standard 2D method [5] and a 3D approach that includes the object consistency module. While a 2D approach (left) can produce cleaner results, it cannot handle multiple depths and perspective distortions (indicated by the red arrows) and stretches the objects. By contrast, both the hybrid approach and its special case preserve the shape of objects. However the 3D method (middle) fails to bring in some parts of the scene (shown by green outline near right edge of image). The hybrid approach (bottom) improves on the 3D case by bringing in more parts of the scene. To further understand how the proposed method can successfully integrate the 2D and 3D information observe the multiple-depth scene in Fig. 11. In nearer depths (orange box) the 3D and hybrid approaches are very similar whereas for intermediate (purple and green boxes) and larger depths (yellow boxes), they produce different results. Clearly, the hybrid approach outperforms for larger depths since the 2D scheme has been activated to bring in those parts of the scene. Both approaches have errors for the specular/textureless objects of the scene.

Fig. 12 shows more results for other natural videos with objects and filled background at multiple depths. The background to be filled ranges from just behind the object (nearly same depth) and gradually moves further out of the camera range. The hybrid approach proves very efficient for background cleaning and naturally integrates the two sources of information to get the best of both methods. Finally, Fig. 13 shows a challenging case with promising results that can inspire future work. This sequence has a large object and camera motion as the camera tries to follow the object(s). Further, most of the depth information is unreliable since the background is far away and the users are constantly moving. The hybrid approach nicely performs background cleaning and avoids using the unreliable depth information. However, there is still room for improvement in filling more parts of the scene with the background.

6. Conclusions

A novel hybrid framework was presented that combines many sources of information (color, depth, stationary and moving object segmentations) for panoramic background fusion and cleaning on RGB and depth data captured by commodity 3D cameras such as Intel RealSense and Google Tango. Compared to traditional 2D or 3D schemes, the hybrid method successfully handles natural scenes with multiple depth layers and camera motions. It can effectively handle errors in depth maps and adaptively work at an object level to generate results with low distortions and object structure artifacts. We compared to a state-of-the-art parallax tolerant stereo stitching method [14, 15] that was designed to handle some of the above problems, however it is unable to handle moving objects and erroneous depth maps, causing ghosting and stretching artifacts in the results. The proposed approach eliminates these artifacts using the object consistency module. In addition stereo or 3D video sequences with large number of frames instead of just two stereo image pairs were considered. Extensive results and comparisons on a variety of real-world scenarios showed that the proposed hybrid pipeline is able to effectively combine multiple sources of information such as color, depth, object and scene segmentation. Different inputs with varying degrees of errors and inconsistencies are adaptively handled and visually pleasing and artifact free results are produced.

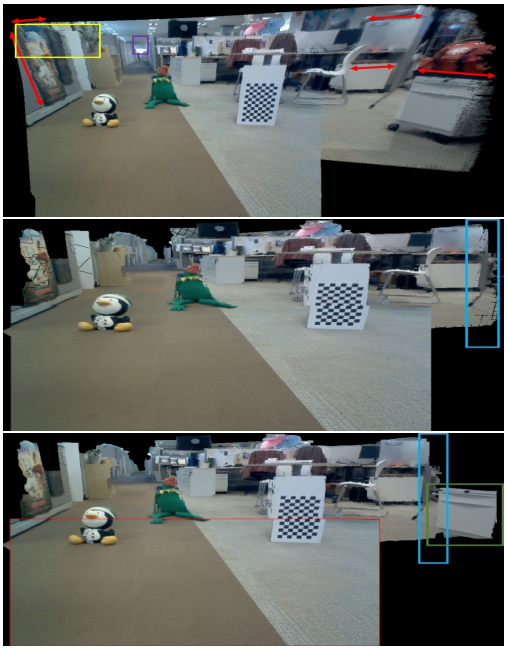

Figure 10. *Comparative results on 'Penguin1' sequence (see Section 5). From top to bottom: standard 2D approach, 3D + object consistency and proposed hybrid pipeline results.*

References

[1] https://www.google.com/atap/project-tango/.

[2] http://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html.

[3] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE PAMI.*, 2012.

[4] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *ACM Transactions on Graphics*, 2004.

Figure 11. *Results from the 'Office' sequence: 3D approach (top), hybrid approach (bottom). Objects at larger depths are preserved in our method.*

Figure 12. *Top row: First frames from two other sequences 'Stanford street-dance' (left) and 'Stanford multiple people' (right). Rows 2,3 Left: Results using pure-3D scheme. Right: Proposed method. See Section 5 for details.*

[5] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *Int'l J. Computer Vision*, 2007.

[6] P. J. Burt and E. H. Adelson. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, 1983.

[7] V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: image and video synthesis using graph cuts. In *ACM Transactions on Graphics*, 2003.

[8] W.-Y. Lin, S. Liu, Y. Matsushita, T.-T. Ng, and L.-F. Cheong. Smoothly varying affine stitching. In *IEEE CVPR*, 2011.

[9] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Intl' Sympos. on Mixed and Aug. Real.*, 2011.

[10] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *ACM Transactions on Graphics*, 2003.

[11] R. F. Salas-Moreno, R. Newcombe, H. Strasdat, P. HJ Kelly, A. J. Davison, et al. Slam++: Simultaneous localisation and mapping at the level of objects. In *IEEE CVPR*, 2013.

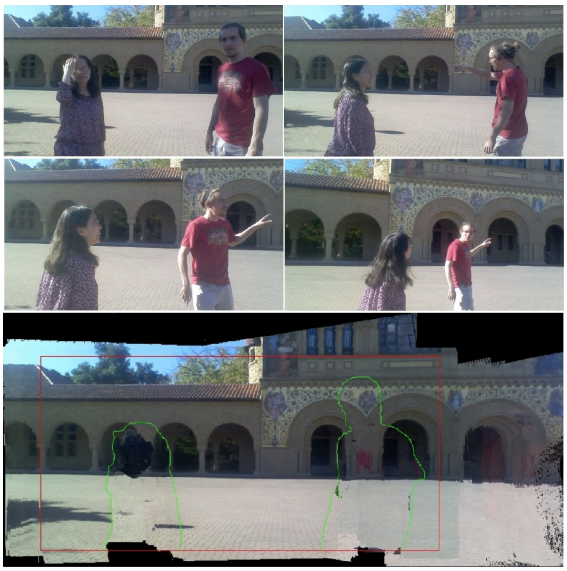

Figure 13. *Rows 1 and 2: Four sample input frames from 'Stanford tour' sequence. Bottom row: Stitched panoramic output with our hybrid scheme which automatically enables its 2D functionality.*

[12] R. Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 2006.

[13] J. Zaragoza, T.-J. Chin, Q.-H. Tran, M. S. Brown, and D. Suter. As-projective-as-possible image stitching with moving DLT. *IEEE PAMI*,, 2014.

[14] F. Zhang and F. Liu. Parallax-tolerant image stitching. In *IEEE CVPR*, 2014.

[15] F. Zhang and F. Liu. Casual stereoscopic panorama stitching. In *IEEE CVPR*, 2015.

[16] G. Somanath, J. Yao and Y. Jiang . A novel framework for fast MRF optimization. In *Electronic Imaging*, 2017.

Author Biography

Christos Bampis received his B.E. in Electrical Engineering from the National Technical University, Athens (2014). He is currently a graduate student in the Laboratory for Image & Video Engineering at the University of Texas at Austin focusing on perceptual image & video quality.

Gowri Somanath recieved her B.E. in Information Science & Enginerring from PES Institute of Technology, India (2006), & her Ph.D. in Computer Science from University of Delaware (2012). She is currently a Research Scientist in Intel Labs focussing on computational photography and computer vision systems and algorithms.

Oscar Nestares received his M.S. (1994) & Ph.D. (1997) in Electrical Engineering from Universidad Politecnica de Madrid. He was a Fulbright Visiting Scholar at Stanford University & consultant at Xerox PARC (1998-2000) & a tenured Research Scientist at the Institute of Optics, Spanish National Research Council (2000-2003). In 2003 he joined Intel Labs focusing on statistical approaches to video enhancement, workload analysis, and computational photography.

Jiajie Yao received his B.S. (2011) and M.S (2014) in Control Theory and Engineering from Zhejiang University. He is currently a Software Engineer in Intel Asia-Pacific R&D Ltd., focusing on image processing and computer vision.