3D Reconstruction of Dynamic Textures in Crowd Sourced Data

Dinghuang Ji, Enrique Dunn, and Jan-Michael Frahm

The University of North Carolina at Chapel Hill
{jdh,dunn,jmf}@cs.unc.edu

Abstract. We propose a framework to automatically build 3D models for scenes containing structures not amenable for photo-consistency based reconstruction due to having dynamic appearance. We analyze the dynamic appearance elements of a given scene by leveraging the imagery contained in Internet image photo-collections and online video sharing websites. Our approach combines large scale crowd sourced SfM techniques with image content segmentation and shape from silhouette techniques to build an iterative framework for 3D shape estimation. The developed system not only enables more complete and robust 3D modeling, but it also enables more realistic visualizations through the identification of dynamic scene elements amenable to dynamic texture mapping. Experiments on crowd sourced image and video datasets illustrate the effectiveness of our automated data-driven approach.

1 Introduction

State of the art crowd sourced 3D reconstruction systems deploy structure from motion (SfM) techniques leveraging large scale imaging redundancy in order to generate photo-realistic models of scenes of interest. The estimated 3D models reliably depict both the shape and appearance of the captured environment under the joint assumptions of shape constancy and appearance congruency, commonly associated with static structures. Accordingly, the attained 3D models are unable to robustly capture dynamic scene elements not in compliance with the aforementioned assumptions. In this work, we strive to estimate more complete and realistic 3D scene representations by addressing the 3D modeling of dynamic scene elements within the context of crowd sourced input imagery.

In our crowd sourced 3D modeling framework, dynamic scene content can only be determined through the observation of visual motion. Nelson and Polana [16] categorized visual motion into three classes: activities, motion events and dynamic (temporal) texture. *Activities*, such as walking or swimming, are defined as motion patterns that are periodic in time; *motion events*, like opening a door, lack temporal or spatial periodicity; *dynamic textures*, i.e. fire, smoke and flowing water, exhibit statistical regularity but have uncertain spatial and temporal extent. Dynamic scenes may contain visual motions in any combination of these three categories. Our work focuses on modeling the 3D shape of scene elements belonging to the dynamic texture category, working under the assumption of a

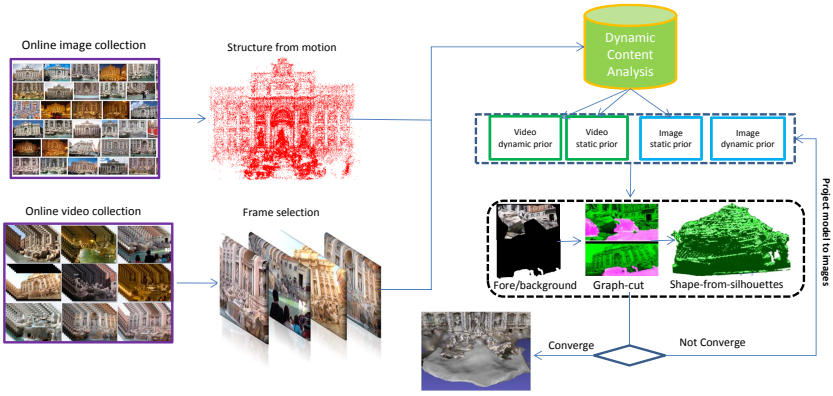

Fig. 1. Workflow overview of the proposed framework.

rigid supporting surface. Moreover, while our framework assumes the geometry of scene elements having time-varying appearance (i.e. such as active billboards or bodies of water) to be approximated by a single surface, our solution is completely data-driven and does not impose geometric or shape priors to perform our estimation.

We briefly summarize the functionality of our processing pipeline. The input data to our framework encompasses both online image and video collections capturing a common scene. We initially leverage photo-collection data to perform sparse reconstruction of the rigid scene elements. Then, video collection data is analyzed to reap video segments amenable for 1) registration to our existing rigid model and 2) coarse identification of dynamic scene elements. We use these coarse estimates, along with the knowledge of our sparse rigid 3D structure, to pose the segmentation of dynamic elements within an image as a global two-label optimization problem. The attained dynamic region masks are subsequently fused through shape-from-silhouette techniques in order to generate an initial 3D shape estimate from the input videos. The preliminary 3D shape is then back projected to the original photo-collection imagery, all image labelings recomputed and then fused to generate an updated 3D shape. This process is iterated until convergence of the output photo-collection imagery segmentation process. Figure 1 depicts an overview of the proposed pipeline.

Our developed system improves upon existing 3D modeling system by increasing the coverage of the generated modeling, mitigating spurious geometry caused by dynamic scene elements and enabling more photo-realistic visualizations through the explicit identification and animation of model surfaces having time varying appearance. The remainder of this document describes the design choices and implementation details of different modules comprising our dynamic scene content modeling pipeline.

2 Related Work

Dense 3D reconstruction of dynamic scenes in uncontrolled environments is a challenging problem for computer vision research. Several systems have been developed for building multiview dynamic outdoor scenes. Jiang et al. [11] and Taneja et. al. [21] propose a probabilistic framework to model outdoor scenes with handheld cameras. Kim et al. [12] design a synchronized portable multiple camera system. These systems rely on a set of pre-calibrated or synchronized cameras, while our method just uses Internet downloaded imagery, which may extensively vary in environment and camera parameters.

Foreground segmentation, which generates the 2D shape of foreground objects, is a critical problem of multiview 3D reconstruction. Many dynamic scene modeling methods only consider controlled environments, where the background is known or can be accurately estimated. Hasler et al. [9] address outdoor scenarios using scene priors, while Ballan et al.[1] limit the reconstruction quality at the billboard level. Taneja et al.[21] propose a method to estimate scene dynamics without making any assumptions on the shape or the motion of elements to be reconstructed. They use the precomputed geometry of the static parts of the scene to transfer the current background appearance across multiply views. Kim et al.[12] propose a multiple view trimap (with foreground, background and unknown labels) propagation algorithm, which allows trimaps to be propagated across multiple views given a small number of manually specified key-frame trimaps. Jiang et al.[11] propose a novel dense depth estimation method, which simultaneously solves bilayer segmentation and depth estimation in a unified energy minimization framework.

Shape from silhouettes is one popular class of methods to estimate shape of scenes from multiple views. Most of these techniques compute the visual hull, which is the maximal volume consistent with a given set of silhouettes. It was first introduced by Baumgart[2], and extensively reviewed by Laurentini[15]. Visual hull is usually in the format of 3D volume, which is a subdivision of space into elementary regions, typically as voxels. Many 3D volume-based visual hull methods, including [6][19][3], are widely used. However, due to camera calibration errors and foreground self-occlusion, traditional shape from silhouette is not robust to noisy input data. Franco et al. [5] propose a sensor fusion method to modify this process and generate more accurate models. In order to address occlusion inference and multi objects modeling, Guan et al.[8] further propose a Bayesian fusion framework.

Scenes with uncontrolled imaging conditions cause many false matches, leading to noisy sparse 3D reconstructions. Tetrahedra-carving-based methods [13][14][23] mitigate this problem by: (1)transforming a dense point cloud into a visibility consistent mesh (2) refine the mesh by geometric and photometric consistency. Jancosek et al.[10] further use visual hull to construct weakly supported surfaces (i.e. road, transparent layers) which are not densely sampled. However, their method does not explore scenarios where dynamic appearance changes are the cause of the reduced support of a given surface.

3 Initial Model Generation

3.1 Static Reconstruction from Photo Collections

The first step in our pipeline is to build a preliminary 3D model of the environment using photo-collection imagery. To this end we perform keyword and location based queries to photo sharing websites such as Flickr & Panoramio. We perform GIST based K-means clustering to attain a reduced set images on which to perform exhaustive geometric verification. We take the largest connected component in the resulting camera graph, consisting of pairwise registered cluster centers, as our initial sparse model and perform intra-cluster geometric verification to densify the camera graph. The final set of registered images is fed to the publicly available VisualSfM module to attain a final sparse reconstruction. The motivation for using VisualSfM is the availability for direct comparison against two input compatible surface reconstruction modules: PMVS2[7] by Furukawa & Ponce and CMPMVS[10] by Jancosek & Pajdla. Once a static sparse model is attained the focus shifts to identifying additional video imagery enabling the identification and modeling of dynamic scene content.

3.2 Coarse Dynamic Textures Priors from Video

Video collections are the natural media to identify and analyze dynamic content. To this end we download videos from YouTube using tag queries of the scenes of interest. Our goal is to identify and extract informative video fragments within our downloaded set of videos. We consider as informative, those video subsequences where the dynamic texture content can be distinguished and reliably correlated with our existing sparse model of the scene's static structure.

Video Frame Registration. We temporally sample each video at a 1/50 ratio to obtain a reduced set of video frames for analysis. For illustration, a set of 500 videos generated little over 80K frame samples. We introduce into the video frame set a random subset of 30% of the registered cameras from the rigid scene modeling. We again perform GIST based clustering on the augmented image set and re-run intra cluster geometric verification to identify registered video frames.

Video Sub-sequence Selection. Given a reduced set of registered video frames we want to select compact frame sub-sequences having reduced camera motion in order to simplify the detection of dynamic scene content. Namely, we compute the HOG descriptor of the frames immediately preceding and following a registered video frame in the original sequence. We count the number of neighboring frames having an NCC value in the range $(0.9, 1)$ w.r.t. the registered frame and keep those sequences having cardinality above a given threshold τ_{seq_len}. We favor such image content based approach instead of pairwise camera motion estimation due to the difficulty in defining suitable capture dependent thresholds (i.e. camera motion, lighting changes, varying zoom, etc.). Discarding fully correlated (i.e. NNC=1) pairwise measurements enables the elimination of duplicates. Moreover, we found measuring the NCC over the HOG descriptors to be robust against abrupt dynamic texture variation as long such changes were

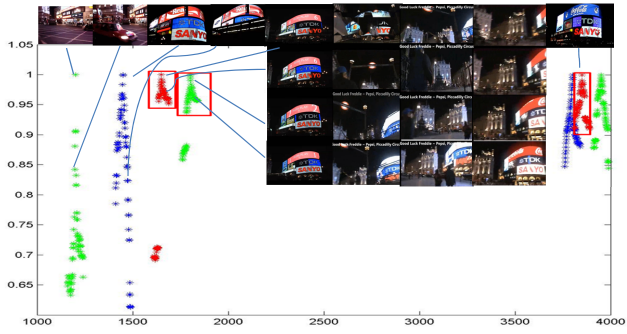

Fig. 2. Keyframe selection for an input video. The plot shows the frame number count vs the NCC similarity of each frame's HOG descriptor. Red boxes indicate selected video fragments centered on sampled frames. Sequences that are not been selected usually have large viewpoint change or severe occlusions(i.e. cars, pedestrians etc.).

restricted to reduced image regions. Figure 2 describes the selection thresholds utilized for subsequence detection.

Barebones Dynamic Texture Estimation. In order to analyze and synthesize dynamic texture from static backgrounds on the selected short video sequences, Soatto et al.[20] and Fitzgibbon [4] propose to model dynamic texture as parametrized auto-regressive model, and compute it with autoregressive moving average process, their works can generate "videotextures" that play forever without repetition. Vidal et al. [22] further work on modeling a scene containing dynamic textures undergoing rigid-body motions, and propose a method to compute both dynamic texture and motion flow of the scene. Since we only want to find the region containing dynamic textures, we deploy basic frame differencing by accumulating the inter-frame pixel intensity differences. We compensate for (the reduced) camera motion by performing RANSAC based homography warping of all sub-sequence frames to the anchor (i.e. registered) video frame. The accumulated difference image is then binarized using non-parametric Otsu thresholding [17]. The attained mask is then modified by a sequence of erosion-dilation-erosion morphological operations with respective window sizes of 2×2 (remove noise), 11×11 (fill holes) and 9×9 (reduce over-grow) for an input image of VGA resolutions. We sort the connected component of the binary output image w.r.t. their area and eliminate all individual components ranked at the bottom 10% of total image area (shown in Figure 3).

3.3 Coarse Static Background Priors from Video Frames

We leverage the dense temporal sampling within a single video sub-sequence in order to estimate a mask for static texture observed on all selected reference video frames. Instead of naively using the complement of the precomputed dynamic texture mask for a given video frame, we strive to deploy a more data-driven approach. To this end we analyze the sparse feature similarity among the

Fig. 3. Dynamic content priors from video fragments. Left to right: (a) Reference frame (b) Accumulated frame differencing (c) Result after post processing.

reference frame and one of its immediate neighbors. We retrieve the set putative SIFT matches previously used for homography based stabilization of the video sequence and perform RANSAC based epipolar geometry estimation. We consider the attained set of inlier image features in the reference videoframe as a sparse sample of the observed static structure. To mitigate spurious dynamic features being registered due to low frequency appearance variations, we exclude from this set any features contained within the regions described by dynamic texture mask. From the final image feature set we compute the concave hull and use the attained 2D polygon as an area-based prior for static scene content (shown in Figure 4).

Fig. 4. Static content prior from video fragments. First and third columns depict SIFT features matches among neighboring frames as red dots. Second and fourth columns depict the concave hull defined by detected features not overlapping with the existing dynamic content prior.

3.4 Graph-cut based dynamic texture refinement

Once a preliminary set of segmentation masks for static and dynamic object regions are attained, they are refined trough a two label (e.g. foreground/background) graph-cut labeling optimization framework. We will denote static structure as

background and dynamic content as foreground. The optimization problem is Graphcut defined as:

$$\min E(f) = \sum_{u \in \mathcal{U}} D_{\mathcal{U}}(f_u) + \sum_{u,v \in \mathcal{N}} V_{u,v}(f_u, f_v) \tag{1}$$

where $f_u, f_v \in \{0, 1\}$ are the labels for pixels u and v, $\mathcal{N}$ is the set of neighboring pixels for u and $\mathcal{U}$ denotes the set of all the pixels with unknown labels. Similarly to the work of Rother et. al. [18], we use a Gaussian mixture model to compute the foreground/background membership probabilities of a pixel. Hence, the smoothness term is defined to be:

$$V_{u,v}(f_u, f_v) = |f_u - f_v| \exp(-\beta(I_u - I_v)^2), \tag{2}$$

where I_u, I_v denote the RGB values of pixels u and v, while $\beta = (2 \left\langle (I_u - I_v)^2 \right\rangle)^{-1}$, for $\langle \cdot \rangle$ denoting the expectation over an image sample. Conversely, the data term is defined as:

$$D_u(f_u) = \log \left(\frac{p(f_u = 1)}{p(f_u = 0)} \right), \tag{3}$$

$$\begin{aligned} p(f_u = 1) &= p(I_u|\lambda_1) = \textstyle\sum_{i=1}^{M} \omega_{i1} g(I_u|\mu_{i1}, \Sigma_{i1}) \\ p(f_u = 0) &= p(I_u|\lambda_0) = \textstyle\sum_{i=1}^{M} \omega_{i0} g(I_u|\mu_{i0}, \Sigma_{i0}) \\ \lambda_{1|0} &= \{\omega_{i1|0}, \mu_{i1|0}, \Sigma_{i1|0}\}, i \in \{1, 2, ..., M\} \end{aligned}$$

and $g(I_u|\mu_i, \Sigma_i)$ belongs to a mixture-of-gaussian model using $M = 3$, and we assume the labels for fore/background are 1/0. Figure 5 exemplifies the result of our graph-cut segmentation.

3.5 Shape from silhouettes

We leverage the output of our graph-cut segmentation module to estimate the 3D visual hull of the dynamic texture through space carving methods. Namely, we utilize the refined dynamic content mask as an object silhouette, along with the corresponding camera poses and calibration estimates, to deploy a 3D fusion method estimating a volumetric shape representation in accordance to the steps described in Algorithm 1.

We first use dynamic appearance silhouettes to determine a visual hull through weighted volume intersection. We observed that segmentation errors occasionally caused overextension of the 3D volume. Our second pass enforces free-space constraints associated with the static background by carving away from 3D volume the silhouettes of the static background. There are two dataset specific thresholds used for space carving: θ_1 and θ_2. θ_1 was empirically set to values from 70% to 90% of total cameras, θ_2 was set from 5% to 15%, roughly (1-θ_1)/2. Higher values of θ_1 slow down convergence by contracting the dynamic texture volume of each iteration, while lower values increase the risk of model over extension. Space carving weight ω_i is set to be 1 in Algorithm 1, in subsection 4.4 we show this value should be adjusted according to camera distribution.

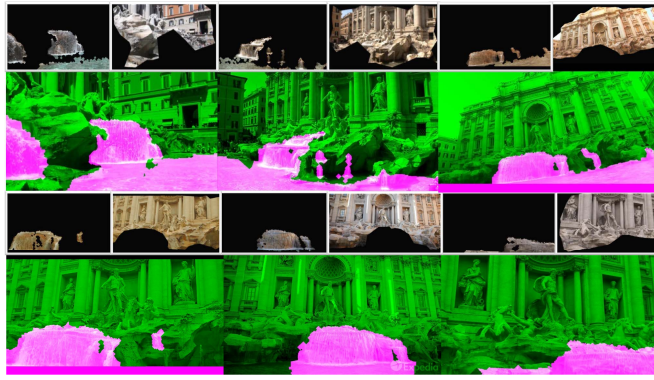

Fig. 5. Graph-cut label refinement. First and third rows depict (alternatively from left to right) single image dynamic and static content priors. Second and fourth rows depict the outputs of the label optimization, where green regions are dynamic textures.

4 Closed Loop 3D Shape Refinement

The preceding section described a video-based approximation of the observed shape of dynamic texture within the scene. The motivation for exclusively using video keyframes until now has been the lack of a mechanism to estimate dynamic texture priors for static images. In this section, we describe an iterative mechanism to effectively transfer the labelings attained from video sequences to the available photo-collection imagery. Such label transferring will enable us to leverage and augmented imagery dataset offering 1) increased robustness through additional redundancy and viewpoint diversity, as well as 2) increased level of detail afforded by larger available imaging resolutions.

4.1 Geometry based Video to Image Label Transfer

In order to transfer dynamic content masks from videos into static images we leverage the estimated preliminary 3D volume. The process is as follows:

1. Generate static background priors for each image.
2. Project the preliminary 3D shape model to all registered images and use its silhouette as a dynamic foreground prior for each image.
3. Execute graph-cut based label optimization for each image.
4. Generate an updated 3D model using the shape from silhouettes module.

Steps 2 to 4 in the above method will iterate until convergence of the dynamic foreground prior mask. Note that in such a framework the static background priors are kept constant while the dynamic texture content is a function of an evolving 3D shape. In general, the preliminary model attained from videos sequences may suffer from variability in viewpoint coverage or be sensitive to errors in our video based dynamic texture segmentation estimates. While the

Algorithm 1: Shape from silhouettes fusion

Input: Sets of camera poses $\{\mathbb{C}_i\}$ and corresponding foreground silhouettes $\{\mathbb{M}_i\}$ and background silhouettes $\{\mathbb{M}'_i\}$, camera weight w_i where $i \in [1, \cdots, N]$, 3D occupancy grid O, threshold θ_1, θ_2

Output: Labeled 3D occupancy grid V

Set all $O(x, y, z) = 0, w_i = 1$
for $i \in [1, N]$ **do**
  for *pixel* $\mathbf{M}_{ij} \in \{\mathbb{M}_i\}$ **do**
    Find all voxels $O_{x,y,z}, \{x, y, z\} \in O_1 \subset O$,$Proj^i(O_1) = M_{ij}$
    $O_1 \leftarrow O_1 + w_i$
$V = Find(\{x, y, z\}|O_{x,y,z} > \theta_1), \{x, y, z\} \in V \subset O$
Set all $V(x, y, z) = 0$
for $i \in [1, N]$ **do**
  for *pixel* $\mathbf{M}'_{ij} \in \{\mathbb{M}'_i\}$ **do**
    Find all voxels $V_{x,y,z}, \{x, y, z\} \in V_1 \subset V$,$Proj^i(V_1) = M'_{ij}$
    $V_1 \leftarrow V_1 + w_i$
$V = Find(\{x, y, z\}|V_{x,y,z} < \theta_2), \{x, y, z\} \in V$
Label voxels in V as occupied.

former may either under-constrain or bias the attained 3D shape, the latter may arbitrarily corrupt the estimate. Both of these challenges are addressed through the additional sampling redundancy afforded by image photo-collections. The remaining challenges consist then in robustly defining static content priors for single images and adapting the shape estimation framework to adequately handle the heterogeneous additional imaging data.

4.2 Mitigating Dynamic Texture in SfM Estimates

The variability in the temporal behavior and extent of dynamic textures may enable its spurious inclusion within SfM estimates. Namely, it is possible for changes in appearance to manifest themselves at time scales larger than those encompassed through short video subsequences or to present periodic behavior that would enable feature correspondence across multiple unsynchronized image. We evaluate the appearance variability of sparse reconstructed features across the imaging dataset to classify them having either persistent or sporadic color.

In principle, static 3D structure with constant appearance should provide consistent color throughout all images observing said structure. Conversely, features with sporadic color are mainly observed from dynamic structures, for example: rocks under the flowing water, flashing letters on a billboard etc. The existence of reconstructed features within a dynamic texture obeys mainly to the transient nature of their appearance. That is, while such appearance is observable at multiple different times, the same structure element may alternatively display appearance independent of the one used for matching.

Moreover, according to Lambert's cosine law, if the colors of a static structure remains constant, the observed pixels are linearly correlated to the intensity of the incoming light, as described by

$$I_D = \mathbf{L} \cdot \mathbf{N} C I_L = C I_L \cos \alpha, \tag{4}$$

where $\mathbf{L}$ and $\mathbf{N}$ are the normalized incoming light direction and the normalized normal for 3D object, C and I_L the color of the model and the intensity of incoming light respectively, making the reflection color I_D a linear function of I_L (with slope $\cos \alpha$). Given that robust features (e.g. SIFT, SURF) enable the robust detection even in the presence of such lighting variation, we can generally expect the color variability of a static feature to comply with such linear behavior. Based on this assumption, we propose a simple method for consistency detection. First we re-project each reconstructed feature to all cameras observing the same structure and record the observed RGB pixel color. Note we re-project to all cameras where the feature falls within the viewing frustum, not just those cameras where the feature was detected. We perform RANSAC based line fitting on the set of measured RGB values to determine the inlier ratio ϵ for a pre-specified distance $d_1 = 0.08$ in the RGB unit color cube. We consider any feature with an estimated inlier ratio below 0.6 to have sporadic color. Figure 6 shows the results running our method on a billboard dataset. Moreover, the set of features classified as having sporadic color will be subsequently used to filter sparse SfM estimates corresponding to static structure.

Fig. 6. Identification of dynamic textures within existing SfM estimates.Top Row: birds-eye and fontal view of estimated sparse structure for Piccadilly Circus. Blue dots are 3D features with persitent color across the dataset. Red dots are 3D features determined to have sporadic color. The bottom row shows sample images in the dataset. We associate color persistance with predominantly linear variation in the RGB space.

4.3 Building a Static Background Prior for Single Images

We leverage the dense spatial sampling within image photo-collections in order to estimate a mask for the static structure observed on all images registered by SfM. In order to achieve as dense as possible sampling of static structure within the image, we retrieve the set of inlier feature matches previously attained by pairwise geometric verification to its closest registered neighbor in GIST-space. We then exclude from this set any features in close proximity to features having sporadic color across the entire dataset. There is a coverage to accuracy trade-off in selecting the pairwise inlier feature set instead of the final reconstructed feature set for each image. In order to mitigate the effect of spurious dynamic texture features, we define a sparse background prior, where each feature location is dilated to define a background mask comprising multiple (possibly overlapping) blob structures. We note the contrast with the area based static prior masks estimated from video (i.e. determined by the concave hull of features). Our rationale is that while the dense spatial sampling of video sequences affords strong spatial correlations, the viewpoint and temporal variability of sparse SfM features provides tightly localized correlations. Moreover, the elimination of features having sporadic color from the static prior enables more robust segmentation by the subsequent graph-cut label refinement.

4.4 Mitigating of Non-uniform Spatial Sampling

In order to generate accurate 3D shape models of dynamic scene elements through space carving methods, wide spatial coverage of cameras is a requisite. In fact, this is the motivation for using photo-collection images. However, the availability of abundant images also presents challenges when said imagery is not uniformly distributed within the scene. Namely, we require a large number of viewing rays tangent to the shape's surface in order for the estimated visual hull to accurately approximate the observed surface. Moreover, our basic shape from silhouettes method will favor the identification of commonly observed image regions. Figure 7 shows the reconstruction of Piccadilly circus using 5800 iconic images (from more than 60,000 images). We can see the camera distribution is not uniform providing scarce coverage of the tangent views of the billboard. In order to compensate for the uncontrolled viewpoint distribution, we deploy a weighting mechanism (Algorithm 2) within our image base shape from silhouettes framework. The procedure reduces the contribution/weight of the cameras having common viewpoint configurations and reduced fields of view. Camera distribution is represented as a histogram of angle values between a reference vector and each of the vectors connecting each camera to the centroid of the 3D initial model.

5 Experiments

We downloaded 4 online datasets from the Internet, with videos attained from Youtube and images from Flickr. The statistics of our systems data associations are presented in Table 1. For all datasets, the set of registered images was

Algorithm 2: camera weighting strategy

Input: A initial model M_0, camera centers $C_i, i \in [1, \cdots, N]$, cameras field-of-view angles $f_i, i \in [1, \cdots, N]$

Output: Space carving weight w_i for each camera

for $i \in [1, \cdots, N]$ **do**
- Direction vector of each camera center $v_i \leftarrow C_i - centroid(M_0)$
- Direction angle of each camera center $a_i \leftarrow \arccos \frac{v_i * v_{N/2}}{norm(v_i) norm(v_{N/2})}$
- $w_i = 1$

Discretize the direction angles into 5 bins histogram centered at $B_j, j \in [1, \cdots, 5]$, with frequency $H_j, j \in [1, \cdots, 5]$

for $i \in [1, \cdots, N]$ **do**
- $idx = find(j | B_j \leq a_i < B_{j+1})$
- $w_i \leftarrow w_i * min(H)/H_{idx}$
- $w_i \leftarrow w_i * min(f)/f_i$

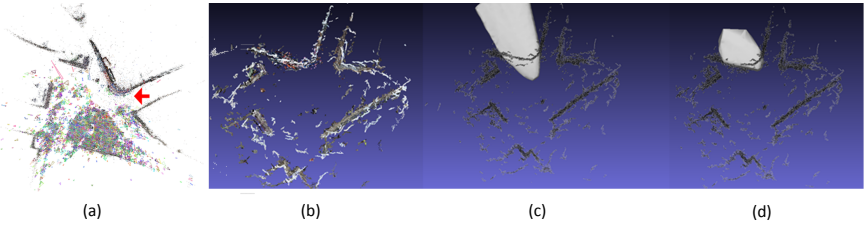

Fig. 7. Mitigation of non uniform spatial sampling. Left to right: (a) Cameras in the red arrow direction are scarse in the SfM model (b) Quasi-dense output from PMVS (c) Dynamic Shape estimation with uniformly weighted carving. The reconstructed 3D volume will be extended towards the camera centroid (d) Shape estimate with weighted carving.

attained using our own SfM implementations, while the final sparse SfM was generated using visualSfM. Figure shows our results combining PMVS quasi-dense model and our dynamic texture shape estimate.

To illustrate the iterative space carving method, we show the segmented estimated visual hull result in each iteration using the Trevi Fountain dataset (Fig. 8). For the the first iteration we use an interaction count ratio (θ_1) of 0.90 and decrease this value by 0.03 each iteration. To ensure convergence of the iteration, we choose a random subset of wide field-of-view images and test their segmentation change in each iteration.

The efficacy of our weighted space carving method for photo collection imagery is illustrated for the Piccadilly Circus Billboard dataset in Figure 7. We can see in the absence of camera contribution weighting, the model will outstretch in the direction of greater camera density. The effect is effectively mitigated by our weighting approach. However, we can still observe slight protrusions w.r.t. the expected surface facade. These are mitigated by a post-processing refinement

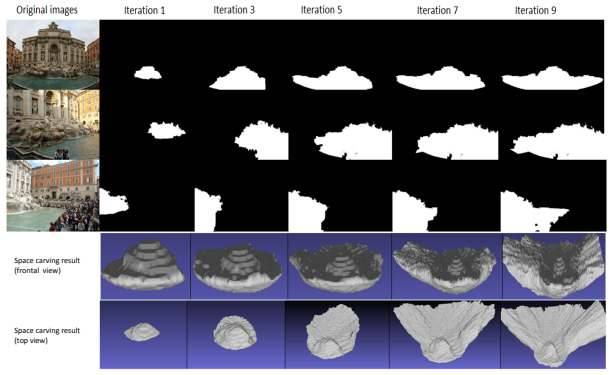

Fig. 8. Evolution of estimated 3D dynamic content in Trevi Fountain model. The video-based model only identified the water motion in the central part of the fountain. Iterative refinement extends the shape to the brim of the fountain. Top rows depict the evolving segmentation mask. Bottom rows depict the evolving 3D shape.

step leveraging the 3D locations of the features determined to have sporadic color (i.e. dynamic texture features reconstructed by SfM) and perform non rigid registration of the final attained dynamic texture shape. We also generate the textured 3D model and compare the results generated by the state-of-the-art method CMPMVS [10] (Fig. 9). For all the experiments, we use the same input dataset for comparison. Each dataset takes approximately 24 hours of processing using both methods.

To illustrate the generality of the proposed framework, we also considered a controlled capture scenario of an indoor scene containing a flat surface with varying illumination. Adapting our method to work with a single input video, instead of crowd sourced data, we were able to generate a 3D approximation of the screen surface of an electronic tablet displaying dynamic texture (shown in Fig. 10). In practice, the inability to attain observations of the dynamic texture of a flat surface from completely oblique views yielded a pice-wise planar 3D surface with a slight outside of plane protrusions. Nevertheless, our attained 3D model was amenable for video texture mapping yielding a realistic animation of the captured video.

Table 1. Composition of our downloaded crowd sourced datasets

Dataset	Videos Downloaded	Keyframes Extracted	Images Downloaded	Images Registered
Trevi Fountain	481	68629	6000	810
Navagio Beach	300	45823	1000	520
Piccadilly Circus Billboard	460	75983	5000	496
Mooney Falls	200	17850	1000	723

Fig. 9. Top two rows: sample dataset imagery, respective outputs for PMVS, CMPMVS and our proposal. Bottom two rows: sample dataset imagery, respective outputs for PMVS and our proposal; CMPMVS failed to generate on the same input data.

Fig. 10. From left to right: sample dataset imagery, respective outputs of PMVS, CMPMVS and our proposed method.

6 Conclusion

We proposed a crowd sourced 3D modeling framework encompassing scene elements having dynamic appearance but constant shape. By leveraging both online video and photo-collections we enable the analysis of scene appearance variability across different time scales and spatial layout. Building upon standard SfM, scene labeling and silhouette fusion modules our system can provide, in a fully automated way, more complete representations of captured landmarks containing dynamic elements, such as bodies of water surfaces and active billboards. Moreover, the segregation of the scene content into static and dynamic elements enables compelling visualizations that incorporate the texture dynamics and effectively "bring 3D models to life".

Acknowledgement This material is based upon work supported by the National Science Foundation under Grant No. IIS-1252921 and No. IIS-1349074. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan GPU used for this research.

References

1. Ballan, L., Brostow, G., Puwein, J., Pollefeys, M.: Unstructured video-based rendering: Interactive exploration of casually captured videos. ACM Transactions on Graphics (2010)
2. Baumgart, B.: Geometric modeling for computer vision. Ph. D. Thesis (Tech. Report AIM-249), Stanford University (1974)
3. Bonet, J.S.D., Viola, P.A.: Roxels: Responsibility weighted 3d volume reconstruction. Proceedings of ICCV 1, 418 (1999)
4. Fitzgibbon, A.W.: Stochastic rigidity: Image registration for nowhere-static scenes. Proceedings. of ICCV p. 662 (2001)
5. Franco, J.S., Boyer, E.: Fusion of multi-view silhouette cues using a space occupancy grid. Proceedings of ICCV 2, 1747 (2005)
6. Furukawa, Y., Ponce, J.: Carved visual hulls for image based modeling. Proceedings of ECCV (2006)
7. Furukawa, Y., Ponce, J.: Towards internet-scale multi-view stereo. Proceedings. of CVPR p. 1434 (2010)
8. Guan, L., Franco, J.S., Pollefey, M.: Multi-object shape estimation and tracking from silhouette cues. Proceedings of CVPR (2008)
9. Hasler, N., Rosenhahn, B., Thormahlen, T., Wand, M., Gall, J., Seidel, H.: Markerless motion capture with unsynchronized moving cameras. Proceedings of CVPR p. 224 (2009)
10. Jancosek, M., Pajdla, T.: Multi-view reconstruction preserving weakly-supported surfaces. Proceedings. of CVPR p. 3121 (2011)
11. Jiang, H., Liu, H., Tan, P., Zhang, G., Bao, H.: 3d reconstruction of dynamic scenes with multiple handheld cameras. Proceedings of ECCV (2012)
12. Kim, H., Sarim, M., Takai, T., Guillemaut, J., Hilton, A.: Dynamic 3d scene reconstruction in outdoor environments. Proceedings of 3DPVT (2010)
13. Labatut, P., Pons, J., Keriven., R.: Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. Proceedings of ICCV p. 1 (2007)
14. Labatut, P., Pons, J., Keriven., R.: Robust and efficient surface reconstruction from range data. Computer Graphics Forum 28, 2275 (2009)
15. Laurentini, A.: The visual hull concept for silhouette-based image understanding. IEEE Trans. on Pattern Analysis and Machine Intelligence 16(2), 150 (1994)
16. Nelson, R., Polana, R.: Qualitative recognition of motion using temporal texture. CVGIP: Image Understanding 56, 78 (1992)
17. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Sys., Man., Cyber. 9(1), 62 (1979)
18. Rother, C., Kolmogorov, V., Blake, A.: Grabcut – interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics 23(3), 309 (2004)
19. Sinha, S.N., Pollefeys, M.: Multi-view reconstruction using photo-consistency and exact silhouette constraints: A maximum-flow formulation. Proceedings of ICCV (2005)
20. Soatto, S., Doretto, G., Wu, Y.N.: Dynamic textures. Proceedings. of ICCV p. 439 (2001)
21. Taneja, A., Ballan, L., Pollefeys, M.: Modeling dynamic scenes recorded with freely moving cameras. Proceedings of ECCV (2010)
22. Vidal, R., Ravich, A.: Optical flow estimation and segmentation of multiple moving dynamic textures. Proceedings. of CVPR p. 516 (2005)

23. Vu, H., Keriven, R., Labatut, P., Pons, J.P.: Towards highresolution large-scale multi-view stereo. Proceedings. of CVPR p. 1430 (2009)