EUROGRAPHICS Workshop on Graphics and Cultural Heritage (2016)
C. E. Catalano and L. De Luca (Editors)

c-Space: Time-evolving 3D models (4D) from heterogeneous distributed video sources

M.Ritz[1], M.Knuth[1], M.Domajnko[1], O.Posniak[1], P.Santos[1], and D.W.Fellner[1,2,3]

[1]Fraunhofer IGD, Germany
[2]TU Darmstadt, Germany
[3]Institut für ComputerGraphik & Wissensvisualisierung, TU Graz, Austria

Abstract

We introduce c-Space, an approach to automated 4D reconstruction of dynamic real world scenes, represented as time-evolving 3D geometry streams, available to everyone. Our novel technique solves the problem of fusing all sources, asynchronously captured from multiple heterogeneous mobile devices around a dynamic scene at a real word location. To this end all captured input is broken down into a massive unordered frame set, sorting the frames along a common time axis, and finally discretizing the ordered frame set into a time-sequence of frame subsets, each subject to photogrammetric 3D reconstruction. The result is a time line of 3D models, each representing a snapshot of the scene evolution in 3D at a specific point in time. Just like a movie is a concatenation of time-discrete frames, representing the evolution of a scene in 2D, the 4D frames reconstructed by c-Space line up to form the captured and dynamically changing 3D geometry of an event over time, thus enabling the user to interact with it in the very same way as with a static 3D model. We do image analysis to automatically maximize the quality of results in the presence of challenging, heterogeneous and asynchronous input sources exhibiting a wide quality spectrum. In addition we show how this technique can be integrated as a 4D reconstruction web service module, available to mobile end-users.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.3]: Digitizing and scanning—Reconstruction; Computer Graphics [I.3.7]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Image-based 3D reconstruction of real world objects nowadays is a widely available process. However, the techniques proposed so far mostly rely on 'patient' 3D objects that are static. In this work we explore ways to perform image based 4D reconstruction of animated objects or entire dynamic scenes and the logistics necessary in order to make the technique usable for crowd-based 4D model acquisition. A 4D model can be understood as time-evolving 3D geometry stream that captures the dynamics of the scene geometry. Capturing the dynamics of physical objects becomes interesting when an ongoing process is to be documented or analyzed, either in real-time, or at specific time intervals over a long period of time. An example for the first scenario of real-time capturing could be manuals for constructing furniture from single parts, where the process of assembly can be viewed in 3D, allowing the viewer to see the entire evolution of the process from any angle necessary to understand the steps involved. Capturing intangible heritage such as dances is another example opening up ways to the viewer to follow actors from arbitrary angles during the performance. The second scenario provides a powerful tool of analysis for science, again in the cultural heritage domain: an excavation site is recorded from many different angles around, during the duration of the dig, with time intervals of minutes or hours between the single images. This allows documentation of the excavation in 3D from the moment of groundbreaking to recovery of single artifacts, helping analysts to accurately assess positions of findings retrospectively from any angle necessary. Also, the entire course of the excavation itself can be analyzed afterwards from arbitrary angles and zoom levels. Due to the necessity of precise time synchronization and correlation of input sources image acquisition for 4D reconstruction is more complex than for the 3D case. Fig. 1 depicts the different capturing modes for static 3D and dynamic 4D reconstruction. The two leftmost illustrations depict classical structure from motion as described and used by Snavely et al. in [SSS06]. A series of images is created taking a video while circulating an object and keeping it constantly in view. These images are then correlated to each other in order to find correspondences in image space. In the end a 3D model can be generated by exploiting disparity between pairs of images. Instead of using a video, single still images of an object can be captured directly, while satisfying the requirement of sufficient overlap between one another for successful correlation between the images. In this way even 3D reconstructions from crowd sourced images of objects can be generated. c-Space takes this idea to the next level: reconstruction is extended by the time dimension. Instead of using single images per camera, video streams are generated to capture

DOI: 10.2312/gch.20161377

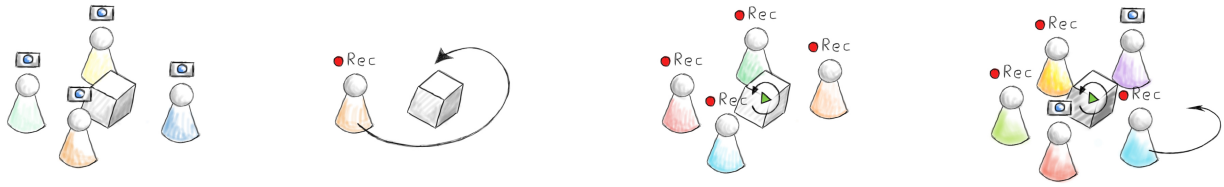

Figure 1: *Reconstruction modes: in contrast to classical photogrammetric reconstruction (left half: multiple still images taken at the same time, and Structure from Motion (SfM) for static objects), 4D reconstruction aims at capturing animated objects. Here, complex capturing situations can occur especially when reconstructing based on heterogeneous data sources from a crowd of people (right half: 4D from videos only, and mixed with still images).*

and reconstruct the evolution of the object or scene over time. This is straight forward and can be relatively easily achieved in a controlled environment with calibrated, time synchronized cameras. However, our scenario is based on a crowd of people taking photos and video material of a scene (rightmost image of Fig. 1). To master this scenario, additional selection, quality-based filtering, time correlation, sorting along a common time axis and rejection of unsuitable input material is required. 4D reconstruction additionally requires precise timing in two ways: first, a global time reference must be enforced among all video and image source devices for tagging recorded footage (down to the frame level) with accurate and comparable time stamps representing the precise moment the image, video or even video frame was captured. Second, correlation of frames contributing to the same 4D time step must be accurate at a low time variance in order to minimize temporal blur of the scene evolution in 3D. Our contributions include:

- Video stream interpreted as frames along time axis vs. as sequence of different static perspectives.
- Combination of different acquisition techniques (still images, video streams and both, static and dynamic, see Fig. 1).
- Techniques for analysis of heterogeneous sets of video streams and still images and their transformation into single scenes subject to 3D reconstruction.
- Techniques to handle moving cameras and align arbitrary geometry coordinate frames within a time-dependent 3D model sequence with the effect of a stable observer perspective throughout the 4D scene evolution.
- Empowering everyone with a mobile device to contribute to the creation of 4D reconstructions, or taking capturing of geometry evolution over time to the professional level using high-end equipment in a calibrated lab setup.

2. Related Work

Several of the typical capturing modes (Fig. 1) can be handled with existing methods. Static object 3D reconstruction based on video can be done on mobile phones in real time as recently demonstrated by Ondruska et al. [OKI15]. However, additional methods are needed to improve tracking of the camera. Static object 3D reconstruction from sets of photos is called Multi-view Stereo (MVS) and has been subject to research for a long time [SSS06, GCS06]. Photogrammetric dynamic 3D point tracking was presented for controlled scenes by Joo et al. [JPS14]. Using the input of communities or crowds adds an additional level of complexity to the reconstruction process, since the scene is no longer captured inside a controlled environment. Therefore, additional methods need to address ways to separate usable images from ones at bad quality, redundant images or even ones negatively affecting reconstruction quality [GSC*07, GAF*10, ARSG12]. Our work adds the time dimension to the 3D reconstruction process. This empowers users to experience the time-dependent evolution of a scene as immersive 4D video and even interact with 4D just like it is possible with 3D geometry, even during the evolution of a dynamic scene. Already in 2004, Goldluecke et al. [GM04] proposed a method of reconstructing the geometry of a dynamic scene, where the time-varying scene is modeled as 3D isosurface in space-time, and intersection with planes of constant time yields the scene geometry at a specific moment in time. A similar approach is used in [BLL16] relying on an indoor multi-camera system for synchronized video capturing, generating the visual hull to reconstruct the volumetric silhouette of a scene at each frame of the video sequence. An approach to space-time multi-view 3D reconstruction was proposed by Oswald et al. [OC13], which generalizes continuous global optimization in multi-view 3D reconstruction to 4D and especially shows robustness for wide-baseline camera setups with unreliable photo consistency measures. The three previously mentioned works aim at including information defining the movement and evolution over time into the reconstruction process of 3D geometry of a dynamic scene which is in accordance to the goal we are pursuing in our work. In contrast to our contribution, however, there is no mention of explicit time synchronization of various sources, especially not for heterogeneous video and image sources with different time spans and frame densities, which we are explicitly dealing with in our approach, which suggests that in these works sets of time-synchronized videos are used that are captured in a well-defined lab setup. Surface- and volume-based shape deformation techniques are applied in [dAST*08] to capture fast movements of persons. Motion constraints are then extracted from video and applied to a laser-scan of the tracked subject, using volumetric deformation to have the 3D model mimic the movement of the actor. Our work, in contrast, solely relies on captured image data to extract geometry, and every moment in our 4D reconstruction captures a moment in the time evolution of the captured scene. We deliberately designed all pre-processing steps of highly heterogeneous input data for the use of the robust photogrammetric Multi-view Stereo technique. One of our main contributions and the core of this work, the timeline discretization of a highly heterogeneous set of input sources - possibly containing both videos and still images of a wide range

of qualities with different time coverages and formats, captured by a large number of independent users with mobile devices - decouples the scene evolution from space by discretization into a set of time-discrete states, each subject to independent photogrammetric 3D reconstruction, while maintaining an iteratively adjusted trade-off between time-coherence and sufficient coverage of multi-view perspectives. As a consequence, our approach supports a wide spectrum of input data, providing the basis for everyone with a mobile device to contribute to the creation of 4D reconstructions, which is the goal of c-Space. Our work is not bound to a specific domain of purpose, the possibilities are only dependent on the quality of the input data. Applications range from entertainment over architecture, science in different areas, to cultural heritage, and many more. In the cultural heritage (CH) domain the nature of scenes, i.e. spatial domains that are to be recorded and reconstructed, is divided into tangible scenes and intangible scenes [Ahm06], even though there is discussion if the distinction might be too binary as the field of dance alone already incorporates a complex, multi-dimensional nature with indissoluble link between both tangible and intangible properties [IB16]. We call tangible CH static due to the rigid nature of objects. Nevertheless, scenes of tangible CH can include objects with time-dependent dynamics, when rigid objects move with respect to one another in a scene. Intangible CH is defined as practices of groups and individuals, where the challenge is movement and changing geometry of possibly the entire scene from one point in time to another during the duration of observation, making this group even more interesting for 4D capture such as the 4D reconstruction of dance performances. The challenge of intangible CH is that object geometries may (and very likely do) change over time. In our work, one 4D frame (3D reconstruction of one point in time of the scene evolution) is independent from any other, so this challenge only comes down to a sufficiently high quality of capturing devices, including resolution and frame rate, and and appropriate capturing setup to minimize blur and noise in images and videos.

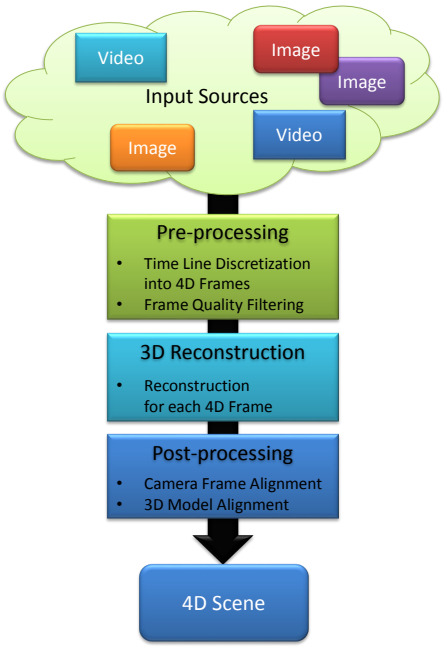

Figure 2: *Overview of the reconstruction core process: Various heterogeneous image sources are first split into single frames, the resulting unordered frame set is then sorted and discretized in time into a sequence of 4D frames. A 4D frame is a set of images of highest quality, contributing to the 3D reconstruction of one geometry representing the evolution of the scene at one discrete moment in time. The series of 4D frames is then reconstructed, frame by frame in parallel, using methods of 3D photogrammetry, leading to a sequence of independent 3D geometry representations of the scene. These independent geometry models are finally aligned with respect to each other using the averaged camera positions determined for each frame.*

3. Concept

We propose a process allowing the use of standard photogrammetric reconstruction for dynamic scenes to be applied on heterogeneous image sources covering a time span. The process results in a series of 3D geometries that, seen as a sequence over time, represent a 4D scene, which is then used like a 3D movie. An overview of the process is given in Fig. 2.

4D input data preparation: All usable footage provided by end-users, captured with arbitrary mobile input devices, is clustered into 'scenes'. Available sensor information from the mobile end-user devices such as GPS position and camera orientation is considered, together with time stamps linked to the imagery according to a global time reference, thereby achieving clustering of information into scenes only containing relevant data. This ensures that scenes are spatially coherent and bounded in the time dimension. Every scene consists of a set of videos and/or still images captured from various perspectives over the duration of the scene evolution. Since c-Space operates on the frame level of the input material, it is transparent to processing whether frames are part of a video frame stream, or originate from single still images. Reconstruction of a scene starts by gathering all relevant material. Every media item comes with tracking information and time stamps necessary for correlating and sorting frames. This information is provided by the c-Space app that is controlling the capturing process for videos and still images on different end-user devices and takes care of a common time by regularly synchronizing itself with a common time server. For later time discretization videos are split into their single frames. This involves extracting and reconstructing still images in full quality for each discrete time step defined by the video stream, with the number of frames depending on the frame rate. Video streams often use run length compression and interpolate between key frames that are inserted every so many frames. Only moving parts in the video stream are recorded in a compressed form, while static (unchanged) parts are redundant information and only fully stored at key frames. The challenge here is to reconstruct every single frame available at maximum quality and completeness. After frame splitting, the scene consists of one single unordered set of frames from different media sources (videos or still images), which from this moment on are treated equally (visualized in Fig. 3, *left* as blue disk). For each frame in this set, all available meta data such as tracking information and, most importantly, the time stamp, are

attached to the respective frame for subsequent correlation along the time axis. The set of frames is yet unordered since it contains frames from all different sources that are possibly interlocking in time. The next step, time line sorting, introduces a strict time sorting by making use of the time stamps attached to each single frame. Time stamps serve as global reference due to the synchronization enforced by the c-Space capturing app in that it maintains a common clock for each device through a central time server prior to capturing. Tracking information includes the GPS position and orientation of the capturing device at the start of the process of capturing a video or still image and additionally for some single frames within a video stream. If no information exists for certain frames originating from videos, information is generated from the starting time stamp of the video and its FPS (frames per second), and by interpolation between information at the closest frames where information is available, such as key frames. One special case of 4D reconstruction is static 3D reconstruction, which can be seen as one discrete time step in the evolution of a 4D model. When providing input data for reconstruction, the user can set a scene to static 3D, in which case all scene media is assumed to have been captured at the same time, as described in Sec. 1 and Fig. 1 *(left half)*, and consequentially, just a single time step is reconstructed as one static 3D model.

Time line sorting: The most important preparatory step for time line discretization, which is the core of 4D reconstruction, is sorting of the unordered frame set. Frame time stamps are used to impose a correlated order among frames along the time axis of the scene evolution to be reconstructed. As Fig. 3 *(left)* indicates, available heterogeneous data sources from various end-users over the time extent of an event for a scene are likely to be unordered in time. At the frame level, different frames typically interlock when projected on the time axis in sorted order due to a large number of time-parallel sequences of frames, making this step vital. The result is a set of frames sorted in ascending order according to their capturing time, each frame originating from one of the input sources (video or still image). The frame distribution for an example scene along the time line versus the different image sources in Fig. 3 *(right)* confirms that the scene dynamics was acquired by 12 videos plus 12 still image frames from individual perspectives at certain times. It can be easily verified that correlation within the set of 24 media sources and alignment on the time axis has been established properly. The 24 rows (12 for videos, 12 for still images) are color coded to allow better identification, with each row representing one individual media source. The visualization contains every input frame, bringing capturing time and originating media source per frame into visual correlation over the common time axis.

Time line discretization: The time range covered by frames between the earliest and latest time stamp needs to be discretized into single frame sets (Fig. 3, *left*), each of them ultimately subject to 3D reconstruction to represent one 4D frame, or discrete time step. A more visual term for the time-discrete nature of the set of frames for one 4D time step is 'bucket', since all contributing frames, captured from different perspectives, are collected in a set of frames (=bucket) which represents one discrete point in time within the time evolution of the scene to be reconstructed. One of the greatest challenges is to meet the optimal trade-off for the bucket size (=time extent of one 4D frame). On one side, this parameter must be large enough to cover a sufficient number of input frames from different sources, and thus perspectives, to allow for successful and robust reconstruction, whereas on the other side, it must be chosen sufficiently small to avoid blur in time evolution, as bucket size corresponds to the time range within which different frames are still considered to contribute to the same state of the scene evolution. The approach to 4D reconstruction proposed uses an iterative scheme to adaptively identify the optimal tradeoff. The number of available frames from distinct sources for each bucket depends both on the intersection set of the time spans covered by each source, corresponding to the number of usable sources over their common time coverage, and on the time density of the single sources, i.e. the number of frames per unit time. Obviously, a bucket cannot contain frames from the same source, as this would at best introduce blur in time evolution of the scene, but not contribute to overlap between different perspectives, and thus to the reconstruction result. Buckets with a frame count too low are rejected since they do not lead to successful photogrammetric 3D reconstruction. The resulting frame rate of the 4D reconstruction is thus variable and depends on the above requirements. In theory, three different perspectives are sufficient for Multi-view Stereo reconstruction (see [GCS06]). In practice, however, a larger lower bound is set to achieve all-around complete and robust reconstruction. For the example in Fig. 3 (*right*), the minimum required number of frames per bucket was set to 12. Details are discussed in Sec. 5. As mentioned above, the iterative approach tries to meet the best tradeoff between completeness and quality of reconstruction on one side, and sharpness of time evolution on the other side. The number of successfully reconstructed 4D frames, and especially the presence of potential streaks of discarded 4D frames, is an indicator as to how well the tradeoff is met. The bucket width, defining the maximum allowed time difference between the latest and the earliest frame within a bucket, is used to control the tradeoff. If buckets are non-overlapping, bucket size corresponds to the time interval between two subsequent buckets. A bucket size too low results in streaks of rejected buckets due to the inability of the discretization algorithm satisfying all constraints for the input frame set. As a response, the discretization process is restarted with an adjusted setting of the parameter. These steps are repeated until both a set of buckets allowing for successful reconstruction, and a 4D stream with sufficiently stable frame rate defined by the number of reconstructed buckets in sequence, is achieved.

The abstract core process of c-Space, time line discretization, is presented by Alg. 1. Precondition is a set of frames F correlating to the same scene, captured from ideally at least 12 different perspectives with homogeneous distribution around the scene and covering the duration of the scene evolution, unordered with respect to capturing time. Each frame is bound to a globally comparable time stamp (with regard to the scene event). The number of elements in a discrete set T is defined here as $|T|$. The time extent of a set T of frames is expressed by $\|T\|_t := max_{\mathbf{t}}(T) - min_{\mathbf{t}}(T)$, with $min_{\mathbf{t}}(T)$ and $max_{\mathbf{t}}(T)$ defining the earliest and latest frame f_t in T according to their time stamps t. The algorithm starts by sorting all frames f_t in F *(line 1)* in ascending order with respect to t, resulting in an ordered set (or list) T of frames. Parameters controlling the algorithm are *minFrames*, the minimum required number

of frames from distinct sources per bucket, and the bucket width or time extent (*maxExtent*). The result of the algorithm is a set of discrete buckets, each of size at most *maxExtent* and containing at least *minFrames* frames, such that as many frames as possible are clustered together in a bucket representing a point in time closest to their time stamps. The subsequent steps can be regarded as a sliding window of size *maxExtent* moving over the time axis. The window is moved over the frames in T, starting from the earliest frame, in the direction of the latest frame, until the window covers the minimum required number of frames (*minFrames*). Then the earliest frame is removed from T and added to the window, which is repeated until the maximum window width *maxExtent* is reached. A new 4D frame is then defined as the set of all frames covered by the window, and the window is moved further to define the next bucket. The positions of buckets, or 4D time stamps, on the time axis are defined by the time distribution and density of the frames, as well as the number and time axis coverage of the sources. The challenge is to satisfy both the minimum required number of frames *minFrames* per bucket and the maximum allowed time span *maxExtent* per bucket at the same time, only by sequentially adding more frames from the time line to and removing earliest frames from the window. In more detail, the sliding window is realized by the loop in *lines 2-14*, which terminates as soon as the ordered set of frames T is empty (as an abstraction, the loop is aborted from any point within as soon as this condition does no longer hold). In every iteration, W is initialized as the empty set *(line 3)*. Until the size of W reaches the minimum number *minFrames* of frames required per bucket, the earliest frame from T is added repetitively to W *(line 5)*, and consequentially removed from T. If the frame to be added originates from the same source as a frame already part of W, the better frame according to an image quality metric as presented by Marichal et al. [MMZ99] is chosen to satisfy the condition of distinct sources per bucket. Adding frames results in a growing time extent of W, a probable exceeded maximum bucket width is thus corrected by repeatedly removing earliest frames from the beginning of W until the maximum bucket width *maxExtent* is again satisfied *(lines 6-8)*. Frames removed are dropped and are not considered in reconstruction, a situation which occurs in the presence of sparse frame distribution of T, either because the number of capturing sources is too low, or due to insufficient frame coverage density per source along the time line. The result of the loop in *lines 4-9* is a frame set W satisfying both *minFrames* and *maxExtent*. The possible resulting leeway in time extent of W is now exploited by the loop in *lines 10-12* in that more frames from T are added until the bucket width limit *maxExtent* is reached. As above, frames added from T are also removed, ensuring single use of frames and termination of the algorithm. Frame set W now satisfies all criteria and is added as new 4D frame, subject to 3D reconstruction, and the algorithm continues with *line 3*, where W is reset to prepare determination of the next frame bucket. Instead of using non-overlapping buckets, possibly sparse bucket distribution can be addressed by allowing a certain bucket overlap, and hence the reuse of a frame for its immediate neighboring buckets. This brings about the benefit that constraints are easier to satisfy, resulting in less 4D frame drops, but with the possible consequence of time evolution blur in the 4D reconstruction. The effect of blur in time evolution of geometry, however, is strongly dependent on the dynamics of the scene, and becomes evident especially whenever the scene changes significantly per unit time, which leads to different frames within the same bucket representing differing evolutionary states of an object. The consequence is that photogrammetry tries to find matching surface points in different perspectives, assuming that the physical counterpart remains in the same position in space. If the positional discrepancy captured in a set of frames is too high, triangulation thus leads to erroneous 3D surface points, i.e., the camera rays of different perspectives that should intersect in the same surface point on the object intersect at an arbitrary distance from the actual surface, or do not intersect at all, which leads to blurry geometry or holes. Our experience shows that a good starting value for bucket time extent is around 50ms. For the special case of static 3D reconstruction (see above), an additional parameter controls the maximum number of images used for scene reconstruction. If the input frame set exceeds this limit, only the best frames according to the quality metric, also used to ensure distinct sources per bucket as described above, are considered.

Algorithm 1 Time Line Discretization

1: $T := sort_{\mathbf{t}}(F)$
2: **while** $|T| > 0$ **do**
3: $\quad W := \emptyset$
4: $\quad$ **while** $|W| < minFrames$) **do**
5: $\quad\quad W := W \cup (f_t := min_{\mathbf{t}}(T)), \quad T := T \setminus f_t$
6: $\quad\quad$ **while** $\|W\|_t > maxExtent$ **do**
7: $\quad\quad\quad W := W \setminus min_{\mathbf{t}}(W)$
8: $\quad\quad$ **end while**
9: $\quad$ **end while**
10: $\quad$ **while** $\|W \cup (f_t := min_{\mathbf{t}}(T))\|_t < maxExtent$ **do**
11: $\quad\quad W := W \cup f_t, \quad T := T \setminus f_t$
12: $\quad$ **end while**
13: $\quad$ *«add W as new 4D frame bucket»*
14: **end while**

4D frame pre-processing: After discretization of the frame set, the quality of the distribution of 4D frames can be assessed. Low frame coverage over the scene duration either due to an insufficient image source count or low time density, or a *bucketSize* setting too small, can have two negative effects. First, the distribution of 4D frames along the time axis is sparse as a consequence of too many buckets being rejected due to the discretization criteria not being met, with the effect of a nonuniform representation of the scene's time evolution. Second, the number of frames within the buckets is just above the requirement, leading to sparse 3D reconstruction. In this case, a feedback loop controlling the parameter *maxExtent* adjusts the bucket size of the discretization as a parameter to counteract the above two negative effects. Bucket size is iteratively increased until a satisfying 4D frame discretization is reached, leading to a representation of the scene's time evolution at higher coverage but lower density (4D frame rate). Using the photogrammetric Multi-view Stereo algorithm, the frame set contained in each bucket results in a 3D geometry model, representing the state of the scene evolution at one specific point in time.

4D frame post-processing: 3D geometry generated by photogrammetry is not aligned, since during 3D reconstruction,

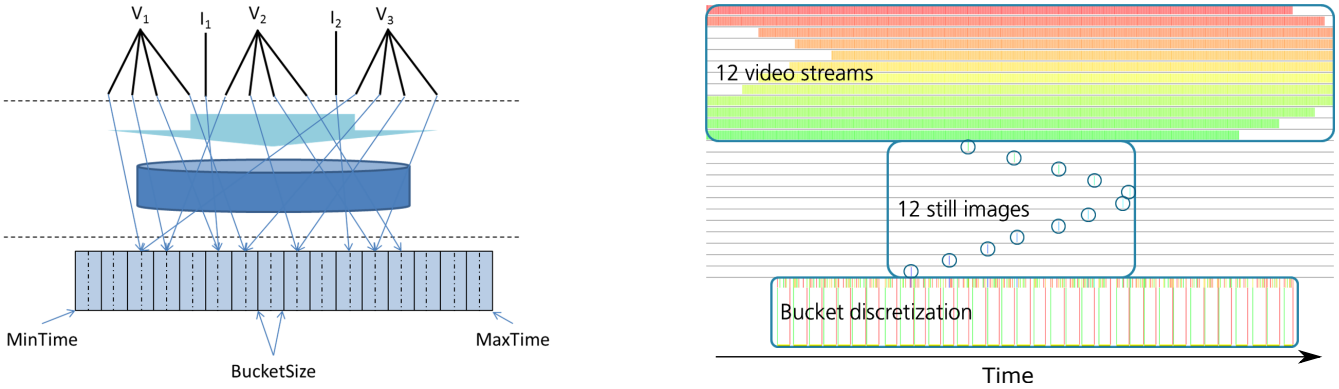

Figure 3: *Left: process chain from heterogeneous input sources (top) over unordered set of frames and time-axis sorting to 4D frame discretization (bottom), right: time-correlated frame distribution for all 12 video (V_i) and 12 still image (I_i) sources of a test scene, visualized as single frames per source (rows), and time discretization into buckets containing selected frames taken from the above sources (below).*

4D frames are processed individually in parallel, independently from the other frames of the 4D model stream, and an arbitrary reference coordinate frame is defined for each. In contrast to the geometries of the 4D frames, the set of known cameras from which the frames of a specific 4D frame were captured, i.e. their positions and orientations in space, stay mostly constant. In order to align all 4D frames throughout the time evolution of a scene our approach makes use of this observation. As a first step, we align the set of camera positions of the current 4D frame to the set of its predecessor, leading to respective transformations that we then apply to the 3D geometry of the respective 4D frame. Since the positions of the cameras are subject to noise and uncertainty, and also to slight movement of the observer guiding the camera, it is not sufficient to find four reference points to describe the orientation from one 4D frame to another. Therefore, we use an approach similar to the deformation technique described by Müller et al. in [MHTG05]. Analogously to the shape that needs to be matched in the work cited, we use the technique to match the current camera setup to the reference (predecessor) setup, and repeat this approach for all 4D frames. As outlined in Fig. 4, our algorithm consists of the following steps:

1. Calculation of offset between current set of camera poses (position and orientation) and reference (predecessor) set of camera poses by determining the offset vector of their centers of mass.
2. Scaling is obtained as the average distance for each camera to the center of mass for the current set in relation to the reference set.
3. Both sets of camera poses now have the same center and scaling, thus the rotation can be obtained by iteratively finding the camera pose with the biggest deviation and rotating the set until it matches the reference set.
4. The transformation for the current camera set, describing the operations necessary to transfer all cameras into the poses of the respective cameras of the reference camera set, is applied to the 3D geometry of the current 4D frame in order to analogously align it to the reference coordinate frame.

These steps are applied to each 4D frame in the sequence, thus aligning all to the reference orientation. The stream of 3D geometries, representing the sequence of discrete evolutionary steps of the scene over time, can then be rendered, e.g., by the c-Space app.

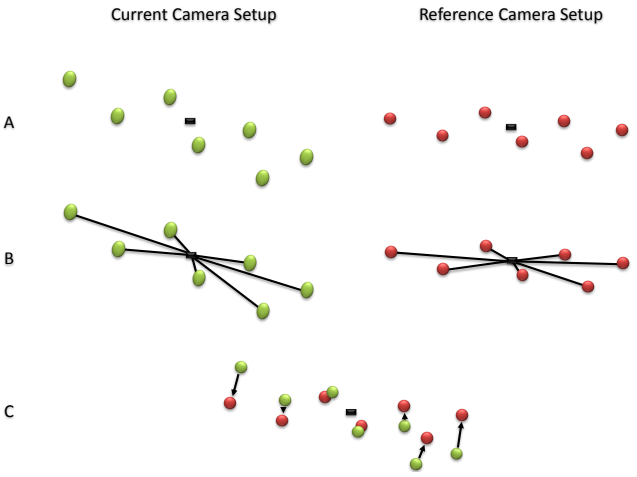

Figure 4: *Determination of 3D geometry alignment transformation between a single 4D frame to a reference 4D frame by aligning their respective camera setups, using a basic shape matching approach: (A) Center of gravity is determined (black box). (B) Average distance of the cameras to the center of gravity for two subsequent frames in relation corresponds to the relative scaling factor between both. (C) After aligning and scaling, the rotation transformation is found that aligns both setups with respect to each other.*

4. Implementation

The 4D reconstruction module is loosely embedded within a web service structure. The c-Space app serves as entry point for video and image footage and interface to the user on smart phones. Also, the app is responsible for providing exact time synchronization among heterogeneous devices of a large number of independent end-users, and to collect sensor information from the devices. The

CAI (Content Access Infrastructure) serves as data base for the footage and allows clustering of the content into virtual scenes, based on the time extent covered, as well as position and orientation of the capturing devices. c-Space is designed for a large number of users, leading to high numbers of scenes to be reconstructed in parallel and massive amount of data. The CAI provides a list of available scenes that were identified as set of relevant media, correlating to the same target object or scene, based on their location, sensor orientation and time information, and having a sufficient number of sources covering the scene evolution. The 4D reconstruction module handles local data synchronization and processing of the input data. For splitting videos into single frames, we use OpenCV (Open Source Computer Vision Library), providing great flexibility in formats and encoding standards. Existing scenes are constantly updated and recomputed as soon as a significant set of new media is available. For the reconstruction of all 4D time steps for a scene as single 3D models, the big advantage of our approach is the use of the standard photogrammetric Multi-view Stereo algorithm used for static scenes. This way the reconstruction for all 4D frames can be performed in parallel. The alignment process (4D frame post-processing, Sec. 3) was implemented as a post-process after completion of all 3D reconstructions. The results show that the algorithm establishes scene continuity, where previously arbitrary coordinate frames had been assumed, leading to a jumping observer perspective. Strong deviations among the camera positions and misalignment can still lead to deviations for single 4D frames, but since the resulting transformations are averaged over all cameras, the impact of single outliers is drastically reduced. The results can be further improved by detecting non-plausible deviations by means of thresholds, but the current results are of sufficient quality. After the alignment processing of all 4D time steps for a scene as single 3D models, each model is compressed and uploaded to the CAI for mobile presentation to the user. Subsequently, the 4D frame sequence is accessed by the post-processing module, which compresses and transforms the data into a 4D stream for efficient rendering on the end-user mobile device.

5. Results and Conclusion

We have presented c-Space, a 4D geometry reconstruction pipeline. The technique allows to capture time-dependent evolution in 3D as immersive and interactive 4D video. Fig. 5 and 6 present 4D reconstruction results for three different dynamic scenes. In theory, three different perspectives are sufficient for Multi-view Stereo reconstruction (see [GCS06]). In practice, however, robust 3D reconstruction at acceptable quality is possible only when satisfying certain conditions during image acquisition. One is the overlap between neighboring perspectives, an overlap of 70-80% is ideal. If the reconstruction is to cover 360° of the scene, we have experienced a number of at least 12 perspectives, placed homogeneously around the scene, to be a minimum requirement. However, it is also possible to only reconstruct a partial arc around a scene, in which case the required minimum number of perspectives can go down to as low as the theoretical number of three. Best results in density and completeness of the geometry reconstructed, however, are reached with a spherical setup of homogeneously distributed cameras, all at the same distance from and facing the center of the object, with the number of cameras high enough to provide sufficient neighboring overlap. The below part of Fig. 3 *(right, below)* shows the result of time discretization consisting of a sequence of buckets. Frames used from the input frame set (above) appear again in the respective bucket (below) to show which frames were actually used. Also, the above requirements are reflected in the graph, in that time intervals with sufficient source coverage lead to buckets subject to successful reconstruction, while others are rejected based on the minimum required number of frames (=different sources/perspectives) per bucket, which for the example was set to 12. Obviously, static pictures as in this document only allow presentation of static 3D renderings for selected time steps. Please watch our 4D rendering videos or try the c-Space app to experience the 4D reconstruction technology. In analogy to the benefit that a video has over a still image, 4D brings an added value over 3D, namely by adding information on the temporal scene evolution and dynamics. The 4D representation of an object or scene as demonstrated by c-Space brings about several benefits: c-Space becomes a spatio-temporally sorted repository of the digital resources available in the area, which end-users can contribute to through a creative crowd sourcing approach in which new localized digital content is created, linked (logically, geographically, and visually), and shared within its surrounding space, and thus fills a gap not covered by any current technology in the market. As examples for specific scenarios, imagine a 4D instruction manual guiding the user through the process of constructing a product from a set of primitive constituents to the final composed state, a furniture item for instance, and all in 3D with nearly unlimited possibilities to view the steps of construction from all around, and even enabling the user to zoom in, remove occluding parts, in parallel to the continuous evolution of the process - compared to current instruction manuals that use a set of discrete states of the process, statically depicted in 2D from one perspective, and abstracting from many important intermediate steps. Other domains where 4D shows nearly unlimited potential are cultural heritage, where dynamic exhibits can be brought closer to the visitor, and research, where scientists

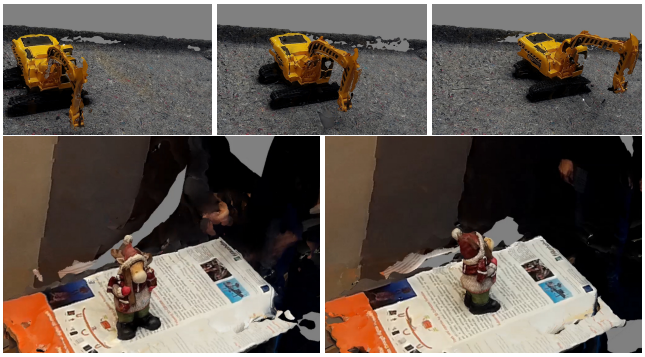

Figure 5: *Results of our 4D reconstruction as 3D renderings of example 4D frames (=discrete scene evolution state at a certain point in time) for two scenes. Top: remote-controlled toy digger in action, bottom: dynamic interaction with figurine. Input images originate from full HD videos (1920x1080) captured by 12 QUMOX cameras with wide angle lenses.*

Figure 6: *Results of our 4D reconstruction as 3D rendering of four example 4D frames for a cultural heritage artifact, being placed upright and assembled. Time sequence is from left to right and from top to bottom. Input images originate from full HD videos (1920x1080) captured by 12 middle class LG smart phones. Noisy background geometry is due to the lack of image features on a homogeneously colored background.*

can profit from the potential of a 4D model to analyze aging of artifacts in 3D, over time. Capturing the dynamics of physical objects becomes interesting when an ongoing process is to be documented or analyzed, either in real-time, or at specific time intervals over a long period of time. Using high-resolution cameras, 4D becomes a powerful tool of analysis for science, again in the cultural heritage domain: an excavation site is recorded from many different angles around, during the duration of the dig, at long time intervals between the single images. This allows documentation of the excavation in 3D from the moment of groundbreaking to recovery of single artifacts, allowing analysts to accurately assess positions of findings retrospectively from any angle necessary. Also, the entire course of the excavation itself can be analyzed afterwards from arbitrary angles and zoom levels. Using high-resolution and high frame rate capturing equipment opens up the possibility of capturing intangible heritage such as dances, thus empowering the viewer to follow actors from arbitrary angles during the performance. Yet another example scenario is capturing dynamic material tests in 4D, opening up the possibility of assessing details about the state of the tested material sample at any point in time in 3D.

6. Acknowledgments

This work was funded in part by FP7-ICT-2013-10 European Grant Agreement No. 611040 - c-Space. More information can be found here: http://www.c-spaceproject.eu/.

References

[Ahm06] AHMAD Y.: The scope and definitions of heritage: From tangible to intangible. *International Journal of Heritage Studies 12*, 3 (2006), 292–300. 3

[ARSG12] ACKERMANN J., RITZ M., STORK A., GOESELE M.: Removing the example from example-based photometric stereo. In *Trends and Topics in Computer Vision*, Kutulakos K., (Ed.), vol. 6554 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 197–210. 2

[BLL16] BLACHE L., LOSCOS C., LUCAS L.: Robust motion flow for mesh tracking of freely moving actors. *The Visual Computer 32*, 2 (2016), 205–216. 2

[dAST*08] DE AGUIAR E., STOLL C., THEOBALT C., AHMED N., SEIDEL H.-P., THRUN S.: Performance capture from sparse multi-view video. In *ACM SIGGRAPH 2008 Papers* (New York, NY, USA, 2008), SIGGRAPH '08, ACM, pp. 98:1–98:10. 2

[GAF*10] GOESELE M., ACKERMANN J., FUHRMANN S., KLOWSKY R., LANGGUTH F., MUCKE P., RITZ M.: Scene reconstruction from community photo collections. *Computer 43*, 6 (2010), 48–53. 2

[GCS06] GOESELE M., CURLESS B., SEITZ S.: Multi-view stereo revisited. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* (2006), vol. 2, pp. 2402–2409. 2, 4, 7

[GM04] GOLDLUECKE B., MAGNOR M.: Space-time isosurface evolution for temporally coherent 3D reconstruction. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on* (June 2004), vol. 1, IEEE, pp. I–350–I–355 Vol.1. 2

[GSC*07] GOESELE M., SNAVELY N., CURLESS B., HOPPE H., SEITZ S. M.: Multi-view stereo for community photo collections. In *Proceedings of 11th International Conference on Computer Vision, ICCV 2007* (2007). 2

[IB16] IACONO V. L., BROWN D. H. K.: Beyond binarism: Exploring a model of living cultural heritage for dance. *Dance Research 34*, 1 (2016), 84–105. 3

[JPS14] JOO H., PARK H. S., SHEIKH Y.: Map visibility estimation for large-scale dynamic 3d reconstruction. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on* (June 2014), pp. 1122–1129. 2

[MHTG05] MÜLLER M., HEIDELBERGER B., TESCHNER M., GROSS M.: Meshless deformations based on shape matching. *ACM Trans. Graph. 24*, 3 (July 2005), 471–478. 6

[MMZ99] MARICHAL X., MA W.-Y., ZHANG H.: Blur determination in the compressed domain using dct information. In *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on* (Oct 1999), vol. 2, pp. 386–390 vol.2. 5

[OC13] OSWALD M., CREMERS D.: A convex relaxation approach to space time multi-view 3d reconstruction. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (2013), pp. 291–298. 2

[OKI15] ONDRUSKA P., KOHLI P., IZADI S.: Mobilefusion: Real-time volumetric surface reconstruction and dense tracking on mobile phones. *Visualization and Computer Graphics, IEEE Transactions on 21*, 11 (Nov 2015), 1251–1258. 2

[SSS06] SNAVELY N., SEITZ S. M., SZELISKI R.: Photo tourism: Exploring photo collections in 3d. In *ACM SIGGRAPH 2006 Papers* (2006), SIGGRAPH '06, pp. 835–846. 1, 2