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

Automatic selection of video frames for path regularization and 3D reconstruction

G. Pavoni[1], M. Dellepiane[1], M. Callieri [1], R. Scopigno [1]

[1]Visual Computing Lab, CNR-ISTI, Pisa, Italy

Abstract

Video sequences can be a valuable source to document the state of objects and sites. They are easy to acquire and they usually ensure a complete coverage of the object of interest.
One of their possible uses is to recover the acquisition path, or the 3D shape of the scene. This can be done by applying structure-from-motion techniques to a representative set of frames extracted from the video. This paper presents an automatic method for the extraction of a predefined number of representative frames that ensures an accurate reconstruction of the sequence path, and possibly enhances the 3D reconstruction of the scene.
The automatic extraction is obtained by analyzing adjacent frames in a starting subset, and adding/removing frames so that the distance *between them remains constant. This ensures the reconstruction of a regularized path and an optimized coverage of all the scene. Finally, more frames are added in the portions of the sequence when more detailed objects are framed. This ensures a better description of the sequence, and a more accurate dense reconstruction.*
The method is automatic, fast and independent from any assumption about the acquired object or the acquisition strategy. It was tested on a variety of different video sequences, showing that a satisfying result can be obtained regardless of the length and quality of the input.

Categories and Subject Descriptors (according to ACM CCS): Three-Dimensional Graphics and Realism [I.3.7]: —Digitization and Image Capture [I.4.1]: —Vision and Scene Understanding [I.2.10]: Video analysis—

1. Introduction

The use of videos for the documentation of artefacts and heritage sites is becoming more and more common. This is mainly due to the fact that Unmanned Aerial Vehicles (UAVs) are now widely available to the community. Despite a lower resolution with respect to photos, video sequences are used not only for plain documentation, but also to obtain a 3D reconstruction.
This 3D reconstruction can be carried out directly from the video sequence, by taking into account the continuity of data (using SLAM or Visual Odometry approaches). Unfortunately, in the case of long videos (more than 2-3 minutes), taken at high resolution (i.e. fullHD), these methods tend to have very long processing times and drift issues leading to error accumulation. Moreover, some assumptions on the video capture (i.e. constant speed, presence of detail) may limit the usability.
An alternative solution adopted by the community is to extract a group of frames from the video and apply Multi-View Stereo approaches. These reconstruction methods were created for set of uncalibrated images, and they are able to handle several hundred images. They are, however, not able to work on *all* the video frames, and a selection of a subset of frames is always necessary.
The choice of the frames to be used is not an easy task: a balance between the amount of images and the quality of coverage is needed, in order to avoid excessive processing times. This is usually obtained in an automatic way by extracting frames at fixed intervals. This approach is fast, and it performs well in the case of controlled acquisition (i.e. fixed speed, predefined and very regular paths), but it is quite inaccurate in the general case.
A manual selection of frames makes it possible to obtain better path and 3D reconstructions, but it is time consuming and relies on the knowledge of the stereo matching algorithm, to understand in advance which are the best frames to extract a regular path and obtain a complete reconstruction.
This paper presents a method to automatically extract an optimal subset of frames from a generic video. The extraction is obtained in an adaptive fashion: the user defines a "budget” of frames, which is the maximum number of frames that should be extracted. Then starting from an initial set of extracted frames, the method refines the set by performing two steps:

- **Regularization of path:** the analysis of the variation in posi-

DOI: 10.2312/gch.20161376

tion and angle of view enables to regularize their distribution in the context of the acquisition path. The method works in a iterative fashion, trying to provide a similar distance value among adjacent frames. The frame budget is preserved, creating a more regular distribution of frames, and a better reconstruction of the acquisition path.
- **Frame set enrichment:** once that the path has been regularized, a second step gives the possibility to change the frame distribution trying to maximize the amount of matches that could lead to an accurate sparse and dense reconstruction. This is achieved by analyzing the amount of information of every frame that is "preserved" by the subsequent one.

The selected frames can be used for the dense 3D reconstruction of the scene. The method has been tested on a variety of video sequences, including free-hand, underwater and UAV videos. The main advantages of the proposed method are:

- *Flexibility:* the frame selection can be performed on any type of video sequence, since no assumption is made about speed, orientation and zoom.
- *Speed:* the method is able to generate a sparse reconstruction of the scene, based on the selected frames, in a comparable time to a subset extracted in an automatic way.
- *Scene independency:* the whole approach does not need a dense reconstruction or an interpretation of the depicted scene. This allows to deal with generic videos, and eventually to use it only to extract a set of representative frames.

This kind of dataset optimization is particularly important when working on the field, a common occurrence for Cultural Heritage UAV surveys, where it is not always possible to have access to servers, and a quick feedback on a freshly-acquired video sequence makes possible an immediate evaluation of its usefulness. Additionally, especially in the case of cultural heritage sites, there are lots of "archival" video sequences, that have not been captured for the purpose of 3D reconstruction. A way to automatically extract meaningful frame subsets could help in re-using existing datasets for a new purpose.

2. Related work

Video sequences are a valuable source of information: several research communities employ them in a variety of applications. Some of them (i.e. tracking of rigid or deformable objects [LF05,SLL10]) may need the extraction of some information about the shape of the scene.

Sparse (and recently even dense) 3D reconstruction is used by SLAM and Visual Odometry approaches [YBHH15] in the context of mobile robotics, to enhance free navigation of unknown environments. Almost all the recent approaches use the principles of Multiple View Geometry [HZ04], where the common features of different frames are used to provide a 3D reconstruction of a scene. Alternatively, the continuity of video data may be used to provide 3D information via optical flow [Zuc02].

More recently, dense 3D reconstruction from videos has been proposed with the goal of having an accurate representation of the scene. This was achieved by using ad-hoc stereo [BFR11] or omnidirectional [ZRTG12] cameras. However, the goal should be to be able to have real-time 3D reconstruction using monocular videos. This was proposed by several recent works, which rely on the fast computation capabilities of graphic cards. They are based on a concurrent segmentation of the scene [KLD*14], on the use of probabilistic models [PFS14], or on the implementation of fast Structure From Motion (SfM) [ND10]. Generally speaking, these reconstruction methods work in a similar fashion w.r.t. depth RGB-D cameras acquisitions: the initial, rough model is refined while additional data are provided by the video.

All of them take into advantage the continuity of the video, and they work under quite strong assumptions: the movement of the camera must be quite *smooth*, and the scene has to exhibit dense geometric and texture details. Hence, it's difficult to apply them when a totally free-hand (or UAV) sequence is provided, or when the characteristics of the acquired scene are not known in advance.

An alternative solution for this type of input is the use of the Multi-View Stereo Matching approaches, which were created to handle uncalibrated set of images. These approaches are based on the principles of SfM: an initial step estimates relative camera positions and provides a sparse reconstruction [SSS06], then a dense reconstruction is calculated [FP10]. Several complete systems [FLM*15, RWFH12, MDDI16] and commercial software [Agi10, Aut12] has been made available, and Multi-View Stereo reconstruction is becoming a standard procedure for several applications, from Cultural Heritage to Forensics.

Nevertheless, using Multi-View Stereo on all the frames of a video is clearly unfeasible, due to the non linear increase of processing time w.r.t. the number of images. Hence, a selection of a subset of frames must be done before the reconstruction. The selection of a set of relevant frames (that was also studied to obtain representative images [CDWS04] of a video) may be crucial to be able to reconstruct the path or the scene. Rachmielowski [RBJC08] proposed a method for the creation of a coarse representation of the scene, showing also the position of some frames of the video. This could guide the user in choosing them for a detailed reconstruction. Huang [HHS08] used self-similarity to extract frames from sequences of human movements. Xiao [XZYW06] performed the same operation using a clustering approach.

The most similar work to our solution was proposed by Rashidi [RDBV13] (and in a similar fashion by Ahmed [ADLH10]): the relevant frames are extracted in an incremental fashion, by trying to obtain a common average *distance* between adjacent ones. Nevertheless, a final global regularization step is needed to better distribute the selected frames. Similarly, Park and Yoon [PY11] first select ”superior features" (features which appear for a long period in a video) and then use them to extract the best key-frames for 3D reconstruction.

Xie [XWB*15] perform a frame selection from acquisition of aerial video sequences, taking advantage of additional information (Flying speed, Global position information, Frame rate) provided by the aerial vehicle.

3. Description of the method

The goal of the proposed method is, given a video and a *frame budget*, to extract an optimal subset of frames, with the aim of being able to regularize the estimated video path and to maximize the coverage and quality of the 3D reconstruction. This is achieved by fulfilling three stages: dataset pre-filtering, path regularization, and dataset enrichment for 3D reconstruction. Each stage is described in the next subsections.

The frame extraction aims at fulfilling a predefined *frame budget*, that is the desired number of frames to be extracted from the video. The reasons of wanting to stick to a frame budget are many: while it may seem intuitive that the more images are used, the better is the 3D reconstruction, this is not completely true. The time required to process a large image dataset increases exponentially, making impractical to process more than 200 images on a "standard" PC. Additionally, it is not true that "more is better", as the small residual errors in the image calibration/orientation, in the case of large photo datasets may produce more noisy reconstruction with respect to smaller photo dataset. Analyzing these problems, Rashidi [RDBV13] numerically evaluated that, in order to obtain an adequate 3D reconstruction from a 25 frames/second video, the number of extracted frames should be between 7% and 10% of the total frames. So, even having access to extremely powerful servers, the "brute force" approach is not effective, and a sensible selection of frames is still the best available option.

When selecting the frames, we have two contrasting needs: obtaining a dataset which is "regular all over” but also "denser where it matters”. We then decided to spend the frame budget in two steps. The proposed method firstly extracts from the video 75% of the *frame budget* and regularize them (Path Regularization), and the remaining 25% will be added in the third stage (dataset enrichment).

Starting from all the frames of the video, we create an initial set of frames by performing an initial extraction that is *regular in time*, using a fixed frame interval which is calculated given the total number of frames and the number of frames to be extracted (75% of the budget).

3.1. Dataset pre-filtering

Given this initial choice of frames, we want first to ensure that each frame is a good candidate for 3D matching. The goal of the pre-filtering phase is two-fold: remove blurred frames and deal with abrupt changes of point of view. The first issue has a strong impact on the quality and robustness of reconstruction, while the second one could impact on the reconstruction of the path. Since no information about the scene acquired and the path is available in advance, SIFT features are extracted on each frame of the first choice set. Their variation will guide the selection of candidates to be removed. The SIFT extraction has no impact on the processing time, since sparse reconstruction will be calculated on the dataset (see later).

The difference between the number of SIFT features of each frame with respect to the previous one is calculated. Given the range of SIFT features available in the dataset (difference between the maximum and minimum number of features found), if the difference is bigger than 40% of the range, the frame is marked as a candidate for removal. The abrupt change in SIFT number usually occurs in two cases: when the frame is blurred (since the number of extracted SIFT decreases), or when the framed scene changes very fast. Every problematic frame detected in this way is removed, and replaced with two frames obtained by regularly subdividing the two neighbor frames intervals. This local increase of frames is aimed at better covering these problematic parts of the path.

It would be possible to detect out-of-focus frames using other image analysis methods, based on contrast factor, or frequency decomposition. However, some of those calculation are not completely scene-independent, may require longer time to compute and, by using SIFT, we are also addressing other irregularities (like sudden changes of view-direction). Moreover, we are using the same kind of data that is used by the structure-from-motion algorithms, making this detection more specific towards the use for 3D reconstruction.

Figure 1 shows two examples of frames selected in the pre-filtering phase. Given the graph of the SIFT found in every extracted frame, the first abrupt change of features number is associated to a fast change of point of view, while the second is related to a single blurred frame. In both cases, the frames are removed and two additional frames are added to "fill the hole”.

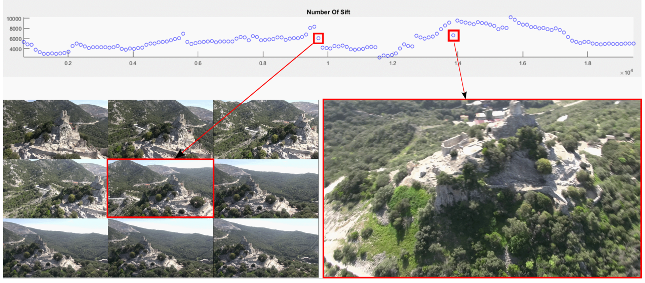

Figure 1: *Two examples of frames selected during the pre-filtering stage. Top: the graph of the number of SIFT extracted for each selected frame. Left: a frame associated to an abrupt change of position and direction of view. Right: a blurred frame.*

Depending on the dataset, other filtering criteria may also be added, to find problematic frames. For example, over/under - exposed frames are a common problem in videos taken using a drone, and can be easily recognized and removed, substituting the bad frames with neighbors that do not present that type of issue.

3.2. Path regularization

Once that the possibly problematic frames have been removed, the second stage aims at regularizing the path, obtaining a sampling which is not *regular in time* but *regular in space*, equalizing the spatial distance and facing angle between frames.

While it is possible to have a drone following a very specific path, with constant linear and angular speed, this is not the general case. Manually controlled flying/underwater drones are subject to accelerations/decelerations, they may stay still in a position, and combine different types of motions at the same time. For these reasons, a set of frames that is regular in time, will not provide a regular coverage of the spatial information contained in the video, resulting in a poor 3D reconstruction.

In order to regularize the set, we need to define a *distance* between the frames, that takes into account both the translation and the rotation of the point of view between the two. To have this, we need to estimate the camera position and orientation in each of the frames

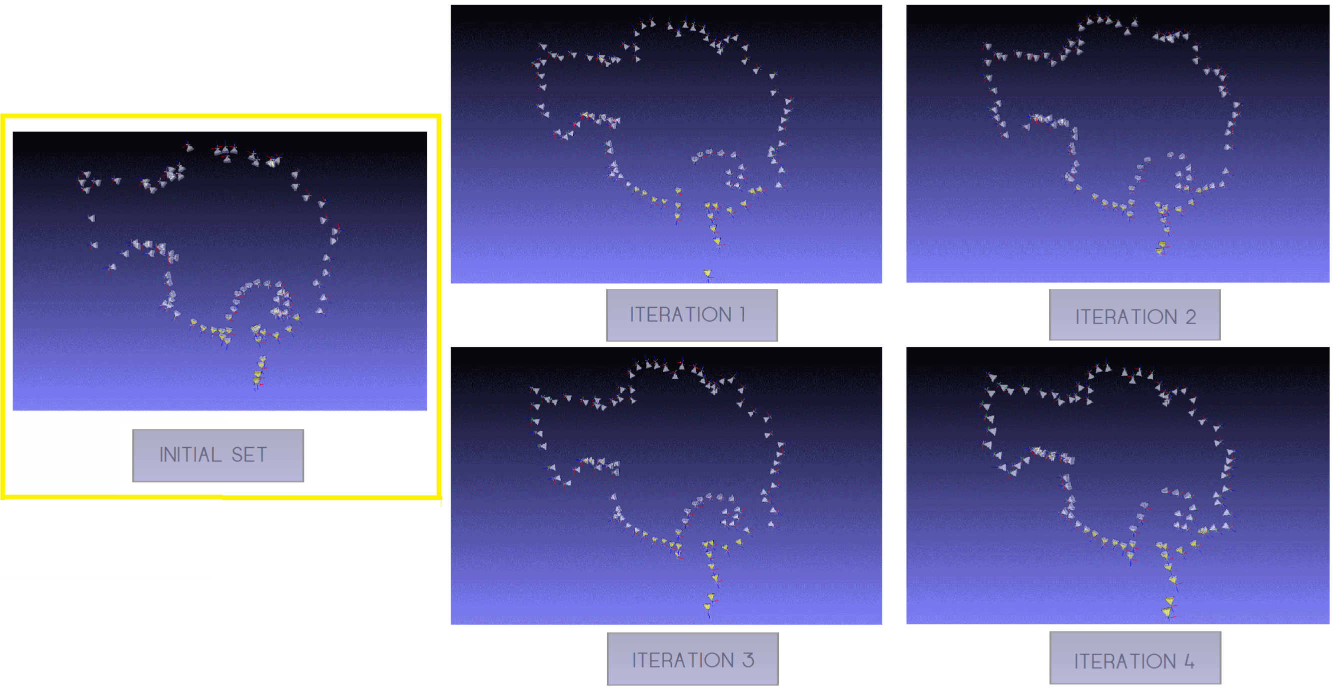

Figure 2: *An example on the first iterations of the Path Regularization stage. The example is the path* Dwarf*, see Figure 8*

of the set: this is exactly the first step of the structure-from-motion reconstruction.
A sparse reconstruction of the path is calculated on the initial frame set, providing an estimation of the path and of the relative position/orientation of adjacent frames (see Figure 2). This reconstruction may be achieved in many ways: by using one of the many commercial software available (like Photoscan) or a scriptable open software (like VisualSFM), or by implementing the basic sparse reconstruction steps (SIFT extractions, matching, camera estimation and bundle adjustment) in MatLab.

From this estimate of the path, the *distance* between subsequent frame is calculated as follows:

$$d = \alpha * d_d + (1 - \alpha) * d_a \qquad (1)$$

where d_d is the Euclidean distance between the estimated view-positions of the cameras associated to the two frames, and d_a is a distance based on the angle a_v between the estimated view-direction of the cameras associated to the two frames. Hence, the defined *distance* value accounts both for changes in position and changes in facing direction.
The distance d_d between the view-positions is linear and straightforward to understand. The angle distance d_a (Equation 2) is slightly more complex, as we used a thresholding scheme because we wanted to highly penalize camera couples with a wider angle difference (and ignore small view-direction changes). This because SIFT matching is extremely vulnerable to changes in view-direction above a certain threshold, and extremely resilient to small angle changes; the values of a_1 and a_2 have thus be set, according to the known strengths and weakness of SIFT matching, respectively at $\pi/18$ and $\pi/6$.

$$d_a = \begin{cases} 0 & a_v \leq a_1 \\ \frac{a_v - a_1}{a_2 - a_1} & a_1 < a_v < a_2 \quad a_v \in [0, \pi) \\ 1 & a_v \geq a_2 \end{cases} \qquad (2)$$

It's possible to change the value of α to assign a different weight to the components of d, if the characteristics of the video do require to give more emphasis to view-position or view-direction distance. In all the examples shown in the paper, α was set to 0.5 .
Once that the relative distance among adjacent frames has been calculated, path regularization can start. The aim of this stage is to try to obtain a constant distance between adjacent frames, trying to maintain the amount of starting frames.
The distance values are ordered, and the values above the 80th percentile distance d_{80} are chosen as candidates for *splitting*. For every frame i_s chosen for *splitting*, new frames between i_s and i_{s-1} are added to the list. The number of new frames is the minimum needed to split the distance associated to i_s into segments which are smaller than the average distance of the whole set. For every frame added during *splitting*, the frame with the lowest distance i_r is chosen for *removal*. A frame is removed only if the distance between i_{r-1} and i_{r+1} is lower than d_{80}. Similarly a consecutive frames set $i_r, ..., i_{r+l}$ can be removed at the same iterative step only if the distance between i_{r-1} and i_{r+l+1} is lower than d_{80}.

For every new frame in a *splitting* operation, a frame in a *removal* operation is needed.

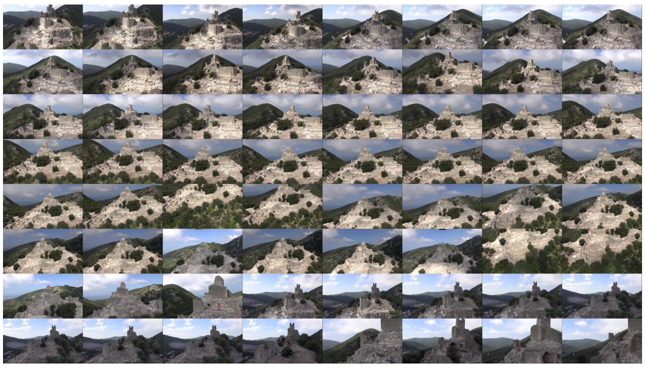

Figure 3: *Some representative frames of the* San Silvestro 2 *sequence*

The regularization is applied in an iterative way: after all the frames above the 80th percentile have been considered, a sparse reconstruction is calculated on the new frame set (see Figure 2). The time needed for this operation is shorter than the first reconstruction, since only the new frames have to be matched with the original ones. After the reconstruction, the *splitting/removal* process is launched again, and the procedure is iterated until no *splitting* or no *removal* is possible.

The convergence is obtained usually after three or four iterations (see Figure 2). The regularization procedure enables to have a proper distribution when the camera moves at different velocities: Figure 4 shows the regularization of the path of sequence *San Silvestro 2* (Figure 3). The frames in the last part of the video, where the UAV had a higher speed, are better sampled after the regularization procedure. In addition, the reconstructed path appears smoother, and some wrongly estimated loops are removed.

3.3. Frame set enrichment

Once that the path has been regularized, the final stage aims at increasing the number of frames in order to have a better coverage of the *object of interest* of the video sequence.

Unfortunately, if no prior information about the path or the scene is known, even when performing dense reconstruction it may be difficult to understand which part of the scene would need more frames. Some effort in this direction has been done in the field of Multi-view reconstruction [DCCS13].

Nevertheless, the proposed method plans to deal with generic videos. Hence, the frame set enrichment is obtained by analyzing the amount of detail that is preserved from one frame to the other.

Given two frames i_i and i_{i+1}, a value

$$R = S_i - M_{i+1} \tag{3}$$

is calculated. S_i is the number of SIFT features with high sharpness and definition (this is obtained by filtering peaks of the DoG scale space that are too small), and $M_i + 1$ is the amount of matches between i_i and i_{i+1}. If R is large, this means that a high number of interesting features was detected in frame i_i, but many have been *lost* in frame i_{i+1}.

Given that the path has been previously regularized (so it's impossible that the features disappear because of the vehicle acceleration), this might be related to two conditions: the video is observing a *feature-rich* object, but it's moving in a direction which is different from the direction of view, or the camera is very near to a *feature-rich* object (the threshold on sharpness of features accounts for this). In both cases, it is necessary to add frames between i_i and i_{i+1} in order to be able to reconstruct (or just to better visualize) further details.

The frame set enrichment inserts new frames between the couples which exhibit the highest R value, until the defined *frame budget* is obtained. Also this stage can be applied iteratively, by adding a subset of the missing frames, re-calculating the sparse reconstruction, and using the new R values to add new frames.

Figure 5 shows the frame set enrichment for *San Silvestro 2* (Figure 3) sequence. In this case, most of the frames where a higher value of R is calculated are related to portions of the path where the UAV was near to the object of interest, or approaching it from a slanted angle.

4. Results

The proposed method has been tested on a variety of types of video, including UAVs, free-hand and underwater. We will show the results obtained in some of them.

This section is divided in two parts: in the first one, the results of the path extraction on some examples are shown, taking into account not only the quality of the paths, but also the processing time needed to obtain them.

The second part focuses on the impact on 3D reconstruction, using also examples where a reference 3D model was available.

4.1. Path extraction

Path extraction is not necessarily just a preliminary step for 3D reconstruction. The estimation of the path and the extraction of a balanced set of frames may be crucial for documentation and analysis.

A perfect example is the sequence that we choose as underwater case-study (see Figure 6). The video was acquired using a ROV (Remotely Operated underwater Vehicle): it is very long (nearly 30 minutes), has a low image resolution (frames are 697x482), and it provides an overview of two adjacent groups of amphorae. It is important to point out that this video has not been shot with the idea of performing a 3D reconstruction.

The vehicle was driven handly, its trajectory is tortuous, it turns over itself portraying the same spot several times, swings or remains perfectly still for minutes. Portrayed objects have very similar shapes and are often occluded by fishes or obscured by aquatic turbidity. The archaeological analysis of the site by simply watching the video is complicated by the total absence of

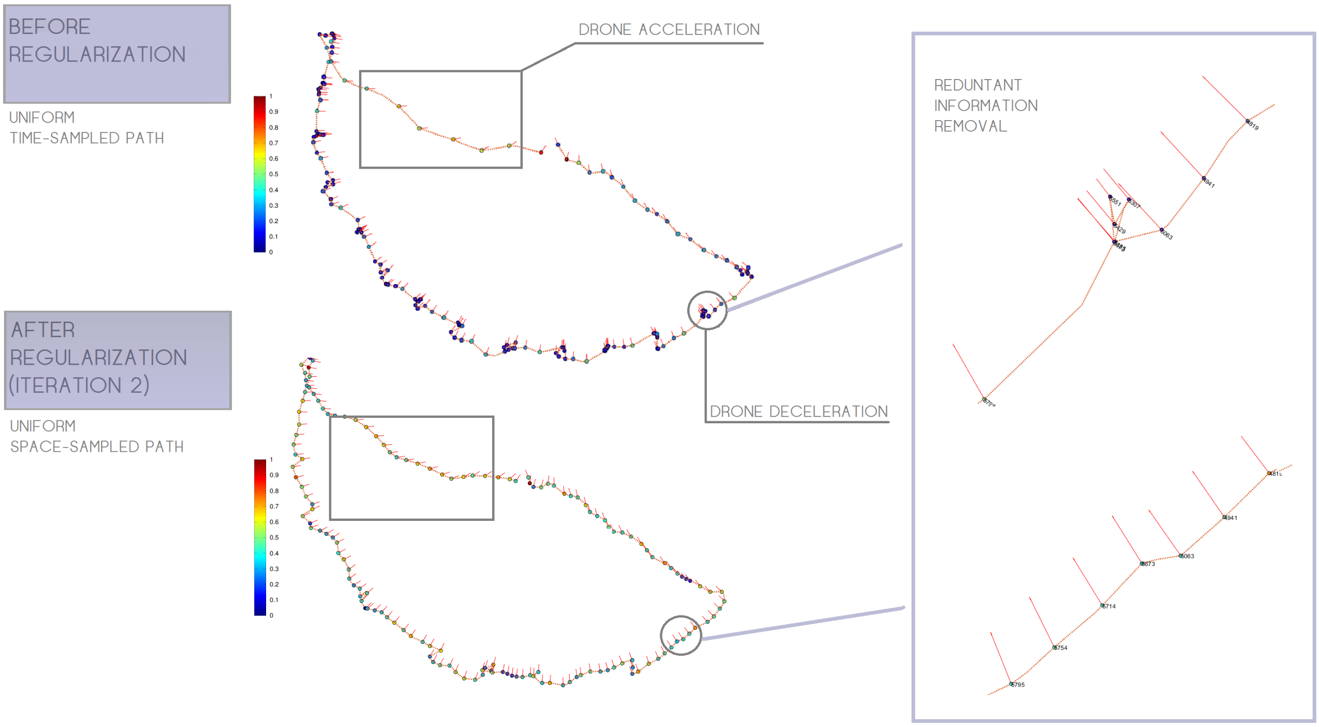

Figure 4: *An example of path regularization, sequence* San Silvestro 2. *Normalized values of distances d_i are mapped into colors from blue to red. Every numbered frame is colored with the associated distance color value respect to its previous. After two iterative step we observe a more dense space sampling for path areas related to drone acceleration. On the contrary a number of frames from oversampled regions with a large amount of dark blue dots (small distance between frames) are deleted. The higher density of frames in some parts of the initial path is related to gusts of wind that forced the drone floating on the same position. (Figure 3).*

reference points, users might lose orientation very easily and it's difficult determinate the exact positioning of amphorae. In this peculiar case provide a meaningful representative photo survey of the site or perform a 3D reconstruction might significantly help archaeologists work.

Nevertheless, the reconstruction of the path is very important, in order to associate the video sequence to the other data (maps, 3D reconstructions, images). The extraction of a good representative set of frames (combining the coverage of the whole path with the detail framing of the object of interest) also makes easier to assess, at a glance, the content of a long video without having to view it completely.

In Figure 7 the reconstruction of the path before and after regularization (using 350 frames) is shown. The most important outcome is the correction of the wrongly estimated path for the first frames, but a better sampling of frames in all the path is also clearly obtained.

Other examples on other datasets have been proposed in previous sections (see Figures 2 and 5). Table 1 shows an overview of the performances of the sparse reconstructions obtained using our method, and using a frame set obtained by extracting the same number of images with a fixed time interval. All the processing was performed on a i7(4 GHz) 8-core PC with 32Gb RAM and a GeForce GTX770 Graphic Card, using VisualSfM tool [Wu11]. Regarding the processing time, the amount of seconds needed for the extraction of best subsets is only slightly bigger. This is due to the fact that the reconstruction has to be performed several time, but the time needed for every iteration is smaller because only a few new images have to be analyzed and matched. Additionally, the amount of matches obtained in the final reconstruction show not only that the path and the coverage are better, but also that the quality of registration is higher. This is shown also by the number of matches that were generated by 3 or more images: these matches are the ones used as a starting point for dense reconstruction. A higher number of matches generated by 3 or more images usually denotes a more "robust" reconstruction.

The main limitation in terms of path analysis and regularization may be in the case of abrupt changes of direction of view. For example, when a rotation of 180 degrees is performed in a short time, it may be possible that the entire path could not be reconstructed, because the camera parameters estimation may fail. This could happen at any stage of the refinement, due to the non coherent behavior of sparse reconstruction approaches even in the case of similar datasets. In this case, a higher weight on the angular component of

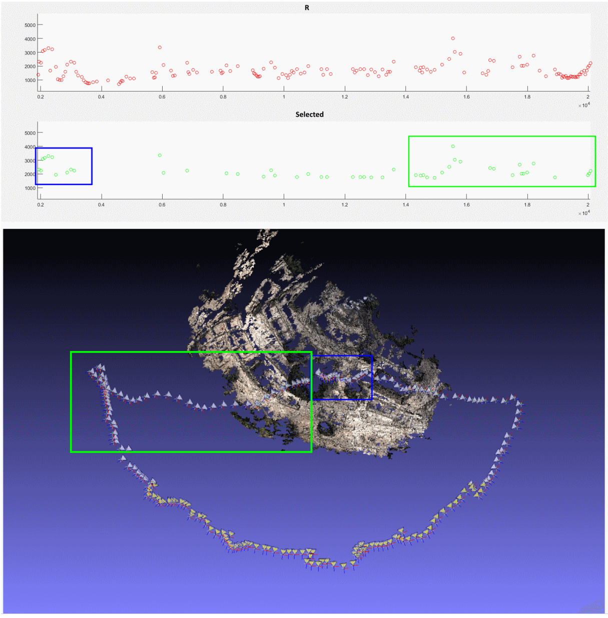

Figure 5: *Frame set enrichment on* San Silvestro 2 *3 sequence. Top: histogram of the R value for all the frames of the path. Middle: histogram of the 50 frames selected for enrichment. Bottom: visualization of the sequence path, with the corresponding areas where most of the enrichment was applied.*

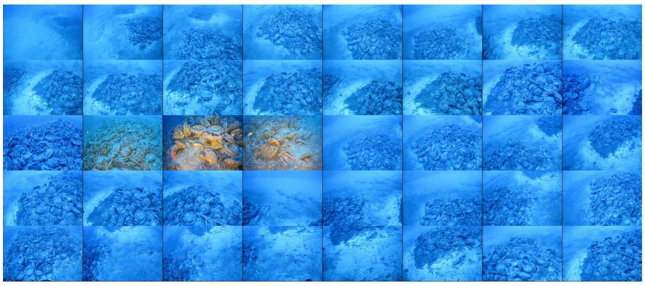

Figure 6: *Some representative frames of the underwater sequence*

Equation 1 may fix the issue. Otherwise, it's also possible to split the path in one or more sub-paths.

4.2. Dense 3D reconstruction

The purpose of the proposed frame extraction method is not only to extract a regularized path, but also to possibly provide the best subset to obtain an accurate dense reconstruction.
We tested the method on several sequences, using also some on which a reference 3D model (obtained via triangulation 3D scanning) was available. In particular, two sequences (*AraPacis* and *Dwarf*, see Figure 8) have a reference 3D model. They were acquired using a smartphone, so the quality of the video is average. The dense reconstruction procedure (obtained using Photoscan [Agi10] tool) was applied on two sets of frames of the same size, one extracted with a fixed time interval, and the other obtained using our procedure.

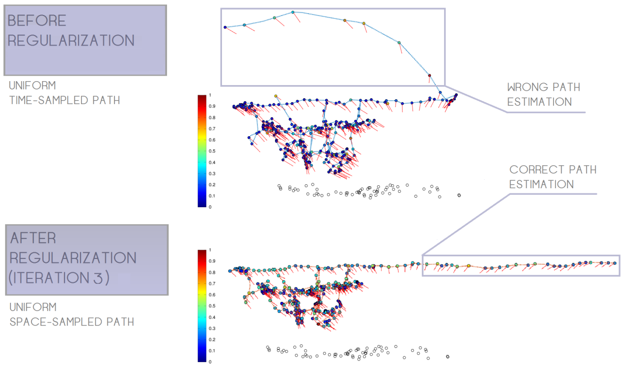
BEFORE REGULARIZATION
UNIFORM TIME-SAMPLED PATH
WRONG PATH ESTIMATION
CORRECT PATH ESTIMATION
AFTER REGULARIZATION (ITERATION 3)
UNIFORM SPACE-SAMPLED PATH

Figure 7: *Reconstruction of the path of the underwater sequence. During its approaching to the archaeological site the ROV traveled with a higher speed. In the initial set an insufficient number of frames are extracted along the first part of the path and the reconstruction algorithm fails in providing a good estimation of the trajectory. After the regularization, with a more uniform frame distribution along the path we obtain the correct ROV trajectory.*

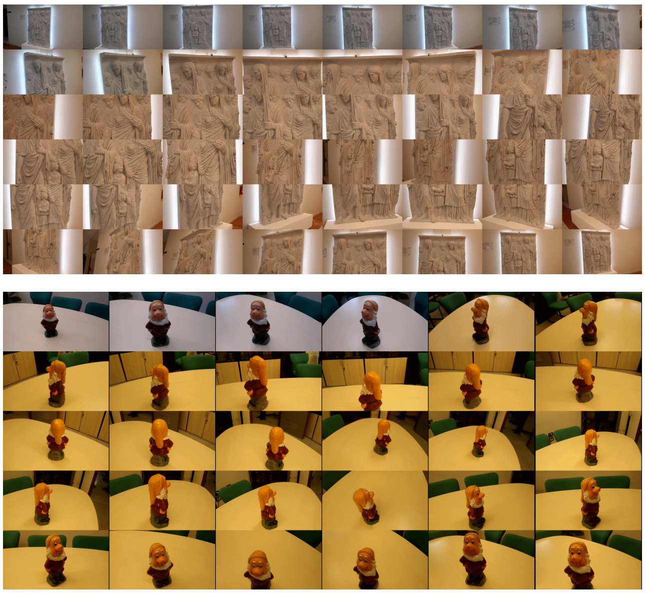

Figure 8: *Some representative frames of the* AraPacis *and* Dwarf *sequences*

The results of the reconstruction are shown in Figure 9 and 10. The *AraPacis* (which has a size of 1.2m x 1.5m) reconstruction provided errors w.r.t. the reference in the order of 15mm maximum. Figure 9 shows that the distribution and amount of error in the cloud obtained with our method is lower and better distributed w.r.t. the reconstruction obtained with the trivial frame extraction method.

In the case of *Dwarf* (29 cm height, Figure 10) reconstruction,

Table 1: *Overview of the performances of our method w.r.t. Fixed Time Interval. he last four columns compare the global performances of the methods.*

	Our method							Fixed Time Interval	
Sequence (Tot. frames)	Starting frames	Proc Time (s)	Regularization Changed frames	Regularization Proc Time (s)	Enrichment Frames/Time(s)	Total Time (s)	Points generated (3+)	Time (s)	Points generated (3+)
Dwarf (1920x1080) 8 3770 frames	100	169	25/13/11/4	45/23/19/7	30/54	*317*	**6507 (4829)**	*259*	**6611 (4464)**
Ara Pacis (1920x1080) 8 3799 frames	100	192	21/30/15/8	43/61/30/17	30/63	*406*	**19483 (14720)**	*343*	**18538 (13529)**
Ventotene (697x482) 29967 frames 6	300	4960	103/46/33	412/191/132	50/230	*5925*	**92429 (56277)**	*5830*	**93998 (57181)**
San Silvestro 1 (1920x1080) 9850 frames 11	150	2120	32/16/10	286/146/91	50/558	*3201*	**99329 (86619)**	*2540*	**92497 (76771)**
San Silvestro 2 (1920x1080) 11075 frames 3	150	2080	47/12	560/105	50/575	*3320*	**65554 (53960)**	*2430*	**54293 (42130)**

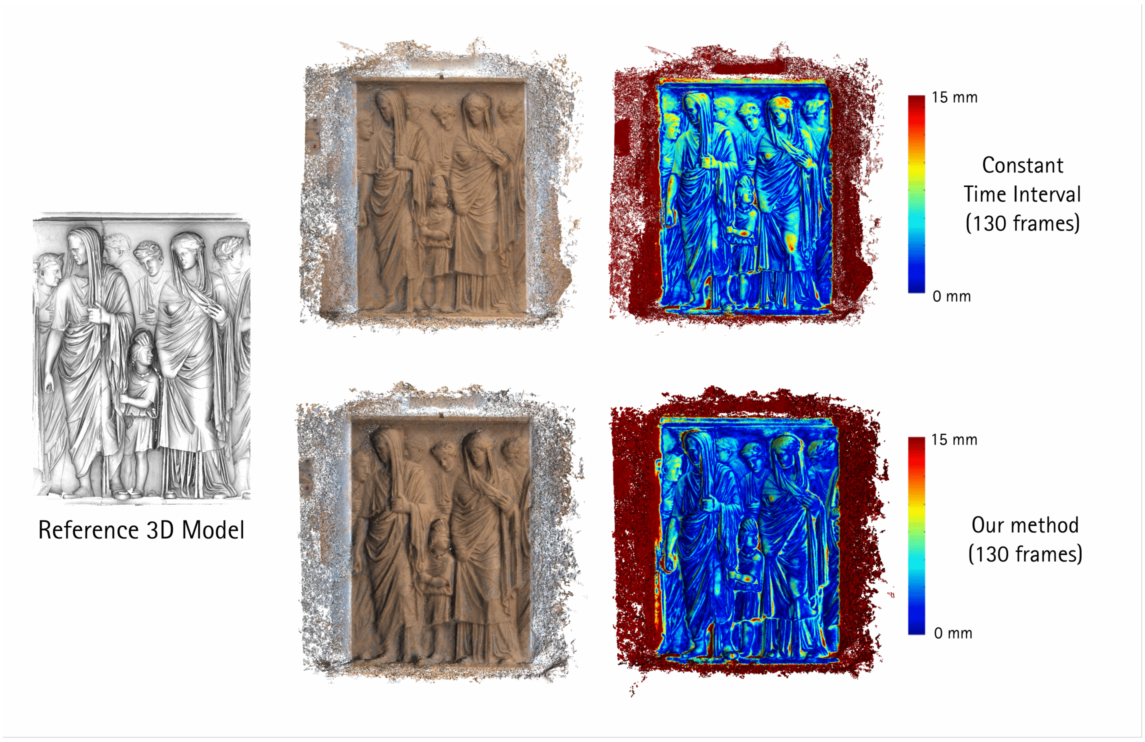

Figure 9: *Quality of dense reconstruction from frame set extracted from video. Each row shows: a snapshot of the reconstructed point cloud, and a mapping of the difference of reconstructed point w.r.t. the reference model*

the quality of reconstructed cloud is not completely satisfying. The main improvement in the use of selected frames is the reconstruction of the lower part of the nose of the statue, that gets lost using the fixed time interval extraction.

Finally, Figure 12 shows an example of dense reconstruction of the San Silvestro 1 (Figure 11) sequence, calculated on a dataset extracted with fixed time interval, and with the frames selected by our method. Our method leads to the reconstruction of a bigger area, and to the removal of some matching errors (i.e. reconstruction of the sky). Additionally, the area of interest is represented by a larger amount of points.

One of the main limitations in terms of 3D reconstruction from videos is represented by the resolution of acquisition, which is usually lower than the one of an uncalibrated photographic dataset. This brings to a limit to the final quality of the reconstructed clouds, and it is one of the reasons why, especially when dealing with UAVs, the users prefer to mount a digital camera and take photos using fixed time intervals. However, the proposed method may be applied also on photographic sets acquired as above, where every image could be treated as a frame, and the selection of the best subset of images may be extracted starting from an arbitrary subset. Experimenting on such types of dataset could lead to clearer results regarding the improvements in dense reconstruction.

5. Conclusions

In this paper we presented an automatic method for the refinement of the extraction of frames from a video, for the purpose of path and

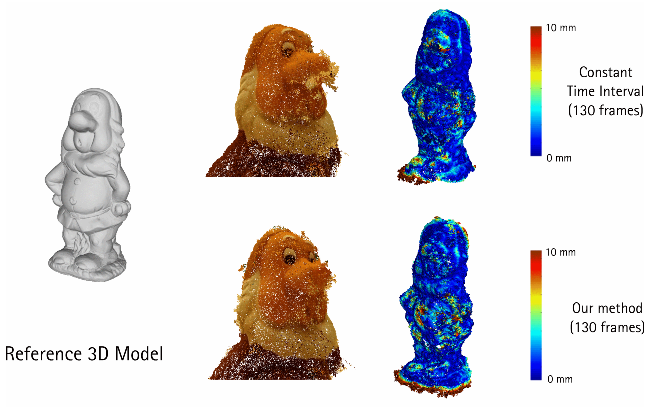

Figure 10: *Quality of dense reconstruction from frame set extracted from video. Each row shows: a snapshot of the reconstructed point cloud, and a mapping of the difference of reconstructed point w.r.t. the reference model*

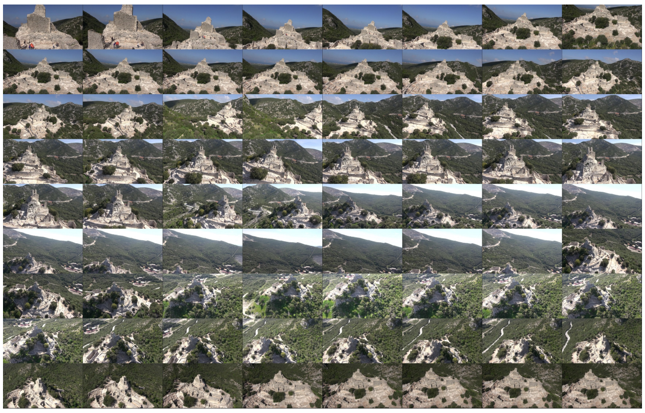

Figure 11: *Some representative frames of the* San Silvestro 1 *sequence*

3D reconstruction. Given a *frame budget*, the method starts from an initial set of frames and it refines the extraction by trying to equalize the distance between adjacent frames. Finally, additional frames are put in the portions of the video where more detail can be extracted from the images.

The method proves to be able to regularize and improve the reconstruction of the video path, and to be able to extract a highly representative set of frames. Additionally, it may help improving the 3D data when dense reconstruction is applied. Given the fact that the path estimation is applied on a smaller number of frames, and that 3D reconstruction is not needed during the frame selection, the method is able to provide the optimal set of frames in a short time and regardless of the typology of video or path.

Some improvements, in addition to the ones related to the limitations described in previous Section, may be devised. Additional control on the trajectory may help in a further regularization of paths: for example, procedures to recognize and eliminate small loops, or small local "hovering" movement of the camera. A way to recognize the main area of interest of the video survey, possibly in an automatic way using the detected SIFTs or some image saliency method, or even exploiting a minimal user input, could help to obtain a more focused extraction.

A major improvement could be the possibility to deal with multiple video sequences of the same scene, in order to have a global frame extraction method. However, this may need the implementation of more "global" matching and camera estimation, with the necessity of longer processing time and hardware resources.

References

[ADLH10] Ahmed M. T., Dailey M. N., Landabaso J. L., Herrero N.: Robust key frame extraction for 3d reconstruction from video streams. In *International Conference on Computer Vision Theory and Applications (VISAPP)* (MAY 2010). 2

[Agi10] Agisoft: Photoscan. http://www.agisoft.com/, 2010. 2, 7

[Aut12] Autodesk: ReCap 360. http://www.autodesk.com/products/recap-360/overview, 2012. 2

[BFR11] Brilakis I., Fathi H., Rashidi A.: Progressive 3d reconstruction of infrastructure with videogrammetry. *Automation in Construction 20*, 7 (2011), 884–895. doi:10.1016/j.autcon.2011.03.005. 2

[CDWS04] Congyan L., De X., Wengang C., Songhe F.: Automatic key-frames extraction to represent a video. In *Signal Processing, 2004. Proceedings. ICSP '04. 2004 7th International Conference on* (Aug 2004), vol. 1, pp. 741–744 vol.1. doi:10.1109/ICOSP.2004.1452769. 2

[DCCS13] Dellepiane M., Cavarretta E., Cignoni P., Scopigno R.: Assisted multi-view stereo reconstruction. In *3DTV-Conference, 2013 International Conference on* (July 2013), pp. 318 – 325. URL: http://vcg.isti.cnr.it/Publications/2013/DCCS13. 5

[FLM*15] Fuhrmann S., Langguth F., Moehrle N., Waechter M., Goesele M.: Mve-an image-based reconstruction environment. *Comput. Graph. 53*, PA (Dec. 2015), 44–53. doi:10.1016/j.cag.2015.09.003. 2

[FP10] Furukawa Y., Ponce J.: Accurate, dense, and robust multiview stereopsis. *IEEE Trans. on Pattern Analysis and Machine Intelligence 32*, 8 (2010), 1362–1376. 2

[HHS08] Huang P., Hilton A., Starck J.: Automatic 3D video summarization: Key frame extraction from Self-Similarity. In *3DPVT '08: Proceedings of the Fourth International Symposium on 3D Data Processing, Visualization and Transmission* (Washington, DC, USA, 2008), IEEE Computer Society. 2

[HZ04] Hartley R. I., Zisserman A.: *Multiple View Geometry in Computer Vision*, second ed. Cambridge University Press, ISBN: 0521540518, 2004. 2

[KLD*14] Kundu A., Li Y., Dellaert F., Li F., Rehg J. M.: *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI*. Springer International Publishing, Cham, 2014, ch. Joint Semantic Segmentation and 3D Reconstruction from Monocular Video, pp. 703–718. doi:10.1007/978-3-319-10599-4_45. 2

[LF05] Lepetit V., Fua P.: Monocular model-based 3d tracking of rigid objects. *Found. Trends. Comput. Graph. Vis. 1*, 1 (Jan. 2005), 1–89. doi:10.1561/0600000001. 2

[MDDI16] Makantasis K., Doulamis A., Doulamis N., Ioannides M.: In the wild image retrieval and clustering for 3d cultural heritage landmarks reconstruction. *Multimedia Tools and Applications 75*, 7 (2016), 3593–3629. doi:10.1007/s11042-014-2191-z. 2

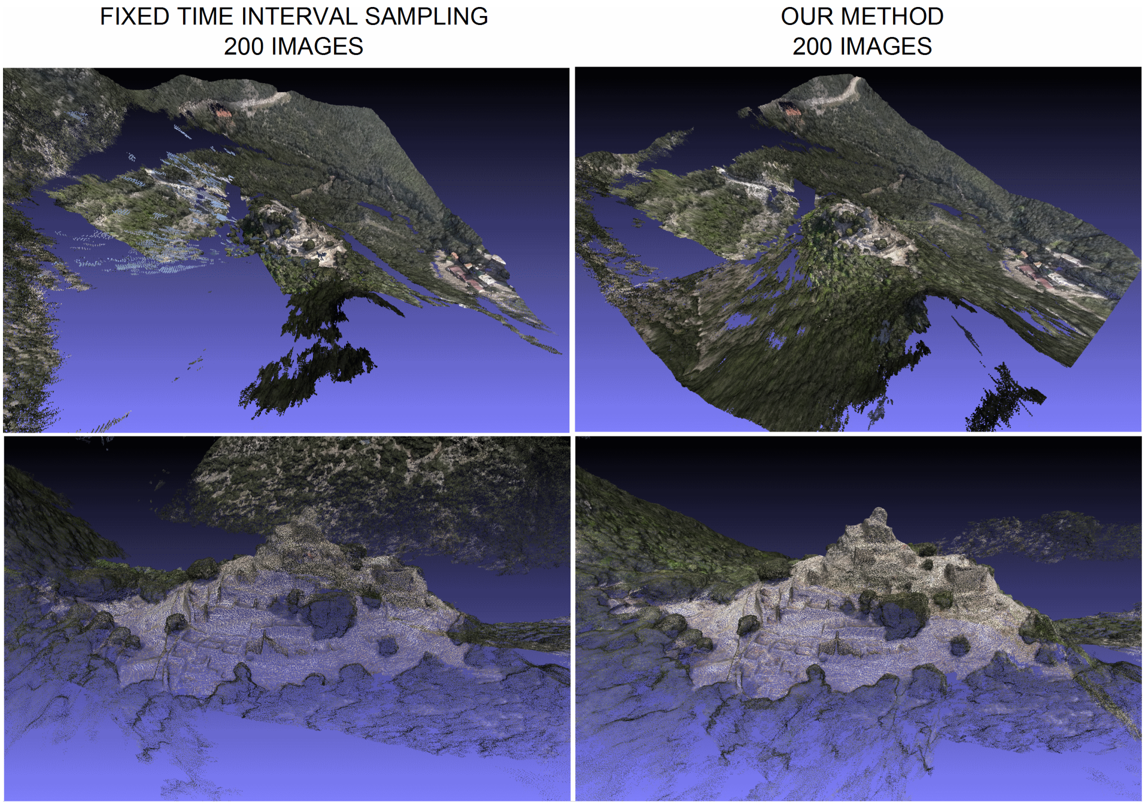

Figure 12: *Comparison of dense reconstruction of San Silvestro 1 sequence, comparison between the fixed time interval and our method. Top row: our method is able to remove wrong matches and reconstruct a bigger area. Bottom row: the points density of the object of interest is higher using the frames extracted by our method.*

[ND10] Newcombe R. A., Davison A. J.: Live dense reconstruction with a single moving camera. In *CVPR* (2010). 2

[PFS14] Pizzoli M., Forster C., Scaramuzza D.: Remode: Probabilistic, monocular dense reconstruction in real time. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (May 2014), pp. 2609–2616. doi:10.1109/ICRA.2014.6907233. 2

[PY11] Park M. G., Yoon K. J.: Optimal key-frame selection for video-based structure-from-motion. *Electronics Letters 47*, 25 (December 2011), 1367–1369. doi:10.1049/el.2011.2674. 2

[RBJC08] Rachmielowski A., Birkbeck N., Jägersand M., Cobzas D.: Realtime visualization of monocular data for 3d reconstruction. In *Computer and Robot Vision, 2008. CRV '08. Canadian Conference on* (May 2008), pp. 196–202. doi:10.1109/CRV.2008.48. 2

[RDBV13] Rashidi A., Dai F., Brilakis I., Vela P.: Optimized selection of key frames for monocular videogrammetric surveying of civil infrastructure. *Adv. Eng. Inform. 27*, 2 (Apr. 2013), 270–282. doi:10.1016/j.aei.2013.01.002. 2, 3

[RWFH12] Rothermel M., Wenzel K., Fritsch D., Haala N.: Sure: Photogrammetricsurface reconstruction from imagery. In *Proceedings LCD Workshop* (Berlin, Germany, 2012). 2

[SLL10] Shen S., Liu Y., Lu W.-S.: Monocular 3d tracking of deformable surfaces using sequential second order cone programming. *Pattern Recogn. 43*, 1 (Jan. 2010), 244–254. doi:10.1016/j.patcog.2009.06.016. 2

[SSS06] Snavely N., Seitz S. M., Szeliski R.: Photo tourism: Exploring photo collections in 3d. In *SIGGRAPH Conference Proceedings* (New York, NY, USA, 2006), ACM Press, pp. 835–846. 2

[Wu11] Wu C.: VisualSFM: A Visual Structure from Motion System. http://ccwu.me/vsfm/doc.html, 2011. 6

[XWB*15] Xie Z., Wan F., Bu Q., Zhou X., Zhang J., Chen S.: Aerial sequential frame decimation for scene reconstruction. In *Information and Automation, 2015 IEEE Int. Conference on* (Aug 2015), pp. 1377–1381. doi:10.1109/ICInfA.2015.7279501. 2

[XZYW06] Xiao J., Zhuang Y., Yang T., Wu F.: *Advances in Computer Graphics: 24th Computer Graphics International Conference, CGI 2006, Hangzhou, China, June 26-28, 2006. Proceedings*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, ch. An Efficient Keyframe Extraction from Motion Capture Data, pp. 494–501. doi:10.1007/11784203_44. 2

[YBHH15] Yousif K., Bab-Hadiashar A., Hoseinnezhad R.: An overview to visual odometry and visual slam: Applications to mobile robotics. *Intelligent Industrial Systems 1*, 4 (2015), 289–311. doi:10.1007/s40903-015-0032-7. 2

[ZRTG12] Zhu M., Ramalingam S., Taguchi Y., Garaas T.: *Computer Vision – ECCV 2012. Workshops and Demonstrations: Florence, Italy, October 7-13, 2012, Proceedings, Part II*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, ch. Monocular Visual Odometry and Dense 3D Reconstruction for On-Road Vehicles, pp. 596–606. doi:10.1007/978-3-642-33868-7_59. 2

[Zuc02] Zucchelli M.: *Optical Flow Based Structure from Motion*. Trita-NA. 2002. URL: https://books.google.it/books?id=a4PsNAAACAAJ. 2