Correspondence Chaining for Enhanced Dense 3D Reconstruction

[1]Oliver Wasenmüller
oliver.wasenmueller@dfki.de

[1]Bernd Krolla
bernd.krolla@dfki.de

[2]Francesco Michielin
michiel4@dei.unipd.it

[1]Didier Stricker
didier.stricker@dfki.de

[1]DFKI GmbH - German Research Center
for Artificial Intelligence
Augmented Vison Department
Trippstadter Str. 122
67663 Kaiserslautern, Germany

[2]University of Padova
Department of Information Engineering
Via Gradenigo, 6/b
35131 Padova, Italy

ABSTRACT

Within the computer vision community, the reconstruction of rigid 3D objects is a well known task in current research. Many existing algorithms provide a dense 3D reconstruction of a rigid object from sequences of 2D images. Commonly, an iterative registration approach is applied for these images, relying on pairwise dense matches between images, which are then triangulated. To minimize redundant and imprecisely reconstructed 3D points, we present and evaluate a new approach, called *Correspondence Chaining*, to fuse existing dense two-view 3D reconstruction algorithms to a multi-view reconstruction, where each 3D point is estimated from multiple images. This leads to an enhanced precision and reduced redundancy. The algorithm is evaluated with three different representative datasets. With Correspondence Chaining the mean error of the reconstructed pointclouds related to ground truth data, acquired with a laser scanner, can be reduced by up to 40%, whereas the root mean square error is even reduced by up to 56%. The reconstructed 3D models contain much less 3D points, while keeping details like fine structures, the file size is reduced by up to 78% and the computation time of the involved parts is decreased by up to 42%.

Keywords

Computer vision, Dense 3D reconstruction, Perspective SfM, Multi-view reconstruction

1 INTRODUCTION

In this work, we consider *3D reconstruction* as the generation of a digital 3D model of a rigid object from a sequence of 2D digital images. This topic, which is focus of intensive research within the computer vision community, deals with the estimation of the relative camera motion and the recovery of the 3D structure of rigid objects from perspective images. It provides applications in many areas such as archeology, virtual reality, human recognition, medical diagnosis, multimedia communication for purposes like documentation, presentation and representation [Cho02].

The topic is especially interesting, since nowadays digital cameras are cheap, widely used and contained in numerous devices such as mobile phones, tablet computers, laptops or even watches. Image-based 3D reconstruction algorithms are able to produce dense and precise 3D models of objects, which can even compete with those produced by laser scanner techniques [Nöl12]. However, these methods demand a highly controlled environment for capturing the images, and are particularly sensitive against difficult lighting conditions. Therefore, in practical daily out-of-lab situations, the 3D reconstruction technology still faces challenging problems.

A widely used approach to 3D reconstruction is to recover the 3D structure from pairs of images of the object, which is known as the two-view reconstruction [Har00, Cho02]. In a two-view reconstruction each 3D point is reconstructed based on only two images, which is the minimum number of images required for

(a) (b) (c)

Figure 1: (a) Reference input image of the *civetta dataset* (sculpture of Gino Cortelazzo [Cor]) and 3D reconstruction results without (b) and with (c) the proposed Correspondence Chaining approach.

(a) (b) (c)

Figure 2: Exemplary input images of the three different datasets. (a) *lion dataset* (27 images, 2808×1872 px). (b) *civetta dataset* (28 images, 2288×1520 px). (c) *temple dataset* (47 images, 640×480 px).

a triangulation. Thus, the triangulation is not robust against imprecise correspondences between pixels. A small baseline between the images results futhermore in a narrow triangulation angle for a two-view reconstruction, which leads to further imprecision of the triangulation. Another problem are redundantly reconstructed 3D points, since identical scene content is reconstructed multiple times for a series of images. These drawbacks can be addressed by a multi-view reconstruction, where each 3D point is triangulated using more than two image points. According to Rumpler at al. [Rum11] a multi-view reconstruction outperforms a two-view reconstruction in terms of precision and redundancy.

In this paper the new Correspondence Chaining algorithm is proposed, which extends existing dense two-view 3D reconstruction algorithms (see Figure 1a) to a multi-view reconstruction (see Figure 1b). The result is a dense 3D model with enhanced precision and reduced redundancy. The algorithm is evaluated based on three representative datasets, which are illustrated by exemplary images in Figure 2, in terms of precision, redundancy, runtime and storage consumption.

The remainder of this paper is organized as follows: Section 2 gives an overview over related work. Section 3 explains the proposed Correspondence Chaining algorithm, which extends existing dense two-view 3D reconstruction algorithms to a multi-view reconstruction. Section 4 evaluates our new method and discusses its results and limitations. The work is concluded in Section 5.

2 RELATED WORK

Rumpler et al. [Rum11] compared in their work two-view against multi-view 3D reconstructions in terms of accuracy and redundancy. According to their results a multi-view reconstruction outperforms a two-view reconstruction by at least one order of magnitude. However, many algorithms in the literature are two-view reconstructions. Thus there is a demand on extending existing two-view reconstruction algorithms to multi-view reconstructions.

When assuming calibrated images the reconstruction quality can also be enhanced with methods like dynamic programming or belief propagation [Sun03]. Furthermore, the depth map fusion approach of Merrell et al. [Mer07] can be applied under this assumption. However, these methods require calibrated images for the dense estimation and we do not want to restrict our method on this assumption.

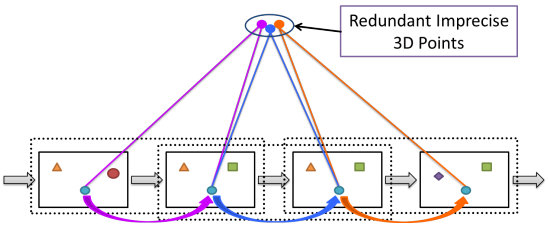
Moulon at al. [Mou12] presented an algorithm to fuse spare correspondences in long uncalibrated image sequences like videos based on the Union-Find algorithm. However, their approach focuses more on low computational complexity than on accuracy. Furthermore only sparse correspondences were considered.

Koch et al. [Koc98] investigated the field of chaining dense two-view correspondences. However, they validate their generated multi-view correspondences exclusively based on statistics. Furthermore the validation depends on the position of the point in the chain. Valid correspondences behind outliers are not considered any more [Koc98].

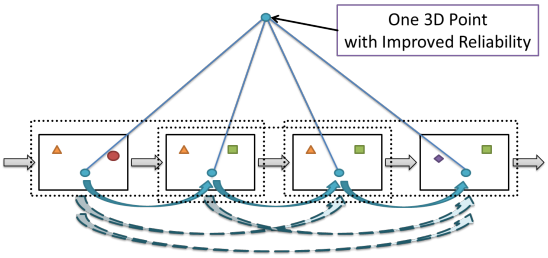
3 METHOD

The proposed Correspondence Chaining algorithm extends existing algorithms for dense two-view reconstructions on uncalibrated images to allow for multi-view reconstruction. To perform two-view reconstruction, any kind of dense correspondence estimation such as optical flow, block matching or patch match methods [Har00] is assumed to be provided. Since common implementations of the listed estimation methods are applied exclusively in a pairwise manner to two neighbored images I_i and I_{i+1} in an image sequence $S = \{I_k | k = 1,2,3,\ldots,n\}$, multiple partial reconstructions of the objects are obtained, when reconstructing rigid objects.

Initial state: In this work, these results of the dense two-view estimation serve as input for the Correspondence Chaining algorithm, opting for a unification of those to one common enhanced 3D reconstruction. The results of the dense estimation are considered to be represented as disparity matrix D_{ij} between image pairs, containing for each pixel $x_i = (u_i, v_i)$ of an image I_i an estimated disparity vector $d_i^{u,v}$ to the neighbored image I_{i+1}. This disparity vector holds the estimated horizontal and vertical offsets between the pixels x_i and x_{i+1}, whereas x_i and x_{i+1} are supposed to represent identical content of the captured object within their images I_i and I_{i+1}. Commonly, the disparity matrix D_{ij} does not contain a mapping between all pixels of the images, since partial occlusions of the scene might occur due to the shifted point of view between images I_i and I_{i+1}. Depending on the chosen object or scene, the dense estimation might furthermore fail, when it comes to the matching of untextured image areas or the formation of view-point dependent specular reflections. The procedure of two-view reconstruction is depicted for one pixel in Figure 3a. Each correspondence $x_i \rightarrow x_{i+1}$ is



(a) Two-View Reconstruction



(b) Multi-View Reconstruction

Figure 3: Two types of reconstruction for four exemplary images.

then triangulated to one 3D point. The result is a dense pointcloud of the captured object.

A major drawback of this pairwise estimation approach is the poor handling of redundant image content, since several input images contain in general identic scene content multiple times. Content of a scene, which is for example contained in m image pairs, will be reconstructed $m-1$ times, leading to redundant 3D points in the resulting reconstruction. While this is neither memory nor runtime efficient, the angle of the triangulation for one correspondence between neighbored images is generally narrow, leading to unreliable triangulation results [Har00]. Since the pairwise triangulations are based on the minimum number of required 2D points, the 3D points are not robust against outlier. Considering more than two 2D points for the triangulation process implies therefore more robust results. To demonstrate this effect, a pointcloud of the *lion dataset* using the two view reconstruction scenario is visualized in Figure 7 (a), whereas the *civetta dataset* can be seen in Figure 8 (a) and the *temple dataset* in Figure 9 (a). Wide parts of the models contain imprecisely reconstructed 3D points, since many points are located in front or behind the surface of the objects, having obviously a wrong position.

Correspondence Chaining: Relying on those results, we propose the Correspondence Chaining algorithm, which extends a dense two-view reconstruction to a multi-view reconstruction, to improve the overall reconstruction quality. Within this algorithm the given dense estimations between image-pairs are chained iteratively to obtain chains with maximum possible length. The iterative procedure of Correspondence Chaining is depicted in Figure 4. The algorithm requires a reference Image, initialized with first image and a target image,

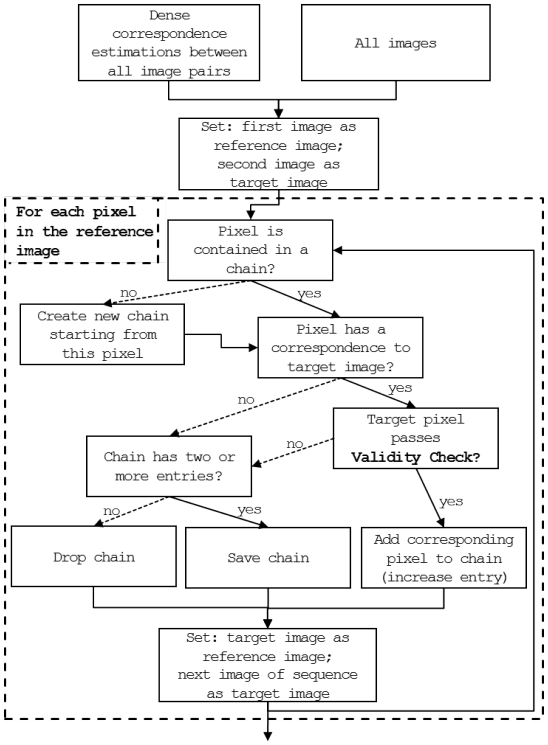
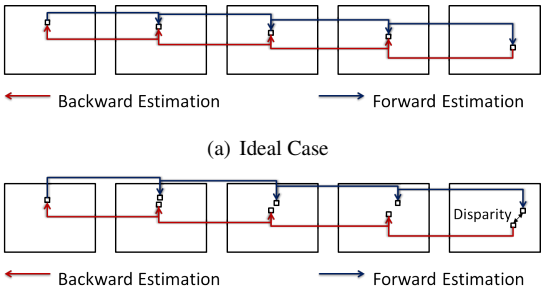


Figure 4: Algorithm of Correspondence Chaining.

initialized with second image and works for every pixel of the reference image.

A check, whether the pixel is already contained in a Correspondence Chain is performed. If this check fails, a new Correspondence Chain, initialized with the actual pixel, is created and the method proceeds with the new Correspondence Chain. Afterwards the existence of a correspondence between the actual pixel and the target image (the next neighboring image), provided by the dense estimation, is checked. If this is the case, this correspondence is validated, since the dense correspondence estimation can be imprecise. For the validity check an extended Round Trip Check (eRTC), which will be detailed subsequently, is applied. The correspondence is only added to the Correspondence Chain C, if it passes the validity check. If the validity check is not passed or if no correspondence was provided, the length of the existing Correspondence Chain is checked: It is rejected, if it has less than two chain links, because two is the minimum number of chain links for a chain. With two or more entries the chain is marked to be completed. For the next iteration step the target image is set as reference image and the next image of the dataset is set as target image.

Iterating over all images of an image sequence S results commonly in long chains of precise correspondences. Afterwards each chain can then be triangulated to one 3D point with improved reliability by applying a multi-view triangulation step, since the generated chains invariably passed the mentioned validity check (eRTC) to eliminate outliers. To further increase the precision



(a) Ideal Case

(b) Real Case

Figure 5: Extended Round Trip Check (eRTC) for five exemplary images.

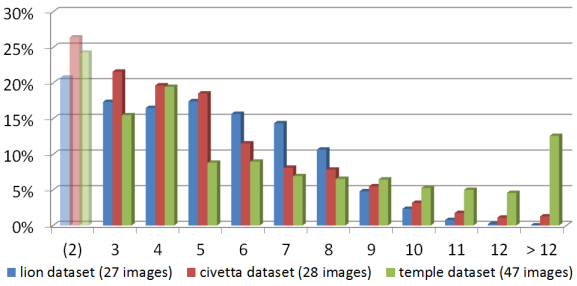


Figure 6: Evaluation of the chain length with Correspondence Chaining. The diagram shows the ratio of a given chain length with respect to the total number of 3D points.

of the method, the Correspondence Chaining algorithm provides a functionality to filter short chains (e.g. with only two or three chain links), avoiding them to affect the resulting 3D pointcloud. Within the current work, chains of length two were removed to not contribute to the reconstruction.

Vadility check: As mentioned before the proposed Correspondence Chaining contains a validity check. This validity check is needed, since the dense estimation is not necessaritly exact and can be imprecise at some points. A precise correspondence is in this work considered to map image content given in image I_i to identical image content given in image I_{i+1}: $x_i \xrightarrow{precise} x_{i+1}$ in pixel precision with $x_i = (u_i, v_i)$ and $x_{i+1} = (u_{i+1}, v_{i+1})$, while for some pixel $x_i = (u_i, v_i)$ an imprecise correspondence is given as $x_i \xrightarrow{imprecise} x'_{i+1}$ with $x'_{i+1} = (u_{i+1} + \Delta_{u_{i+1}}, v_{i+1} + \Delta_{v_{i+1}})$. With values $\Delta_{u_{i+1}}, \Delta_{v_{i+1}} > 0$ image content is not matched correctly anymore, whereas Δ holds typically small values in the order of few pixels such as $\Delta = [-2, 2]$, whereas extreme outliers ($\|\Delta\| \gg 2$) are also possible. In the two-view triangulation $\Delta_{u_{i+1}}$ and $\Delta_{v_{i+1}}$ lead of course to imprecise results, but – as $\Delta_{u_{i+1}}$ and $\Delta_{v_{i+1}}$ are small – their impact is limited. However, in the Correspondence Chaining the deviations $\Delta_{u_{i+1}}$ and $\Delta_{v_{i+1}}$ can lead to problems. While chaining the correspondences, the small deviations $\Delta_{u_{i+1}}$ and $\Delta_{v_{i+1}}$ can accumulate. For example, for a chain with ten chain links and a constant deviation $\Delta = 2$, the estimated chain sums up an error of 20 pixels. To overcome this issue of accumulating errors, a validity check is applied. This check detects imprecision in the dense estimation at a given position by verifying, whether a new chain link together with the already existing chain is plausible.

For Correspondence Chaining we propose the new extended Round Trip Check (eRTC) as a validity check. The eRTC verifies a new chain link c_{n+1} together with the already existing chain $C = \{c_1, \ldots, c_n\}$ based on a forward and backward dense estimation, with backward estimation as estimation from I_{i+1} to I_i. The dense estimation with Correspondence Chaining can estimate a correspondence on a set of images for a pixel $x_n = (u_n, v_n)$ in the last image ($i = n$) to the pixel $x_1 = (u_1, v_1)$ in the first image ($i = 1$). The result is the correspondence $x_n \to x_1$. In the ideal case (see Figure 5a) this correspondence can also be estimated in the opposite direction from the pixel $x_1 = (u_1, v_1)$ in the first image ($i = 1$) to the pixel $x_n = (u_n, v_n)$ in the last image ($i = n$). The result is then the correspondence $x_1 \leftrightarrow x_n$. However, in praxis the forward and backward dense estimation must not be a bijection (see Figure 5 (b). In general holds

$$x_n = (u, v) \to x_1 \quad \Longrightarrow \quad x_1 \to x'_n = (u + a, v + b), \quad (1)$$

where a and b are typically small values (e.g. 1 - 2 pixel). While chaining the correspondences, the small errors a and b accumulate. A disparity d between the pixel, where the estimation started, and the pixel, where the estimation in the opposite direction ended, occurs and this disparity d can be used as a quality measure. If the maximal disparity d is below a given threshold (e.g. two pixels), the new chain link passes the validity check, otherwise not.

4 EVALUATION AND RESULTS

In this Section the Correspondence Chaining algorithm is evaluated. First is the proposed chaining approach investigated by inspecting the resulting chain length at different positions within the 3D model. Furthermore are the runtime and the storage consumption analysed. To verify the precision of the reconstructed pointclouds, a comparison against ground truth data is performed. Finally, meshes are created from the pointclouds to inspect the details in the reconstructed models.

Since the Correspondence Chaining algorithm extends an existing two-view reconstruction to a multi-view reconstruction, an exemplary two-view reconstruction algorithm is required for the evaluation. In this paper we used a estimation method provided by Sony.

Correspondence Chaining: Table 1 and Figure 6 show the number of generated chains for the *lion dataset* listed with respect to their length. The longest chains are based on more than ten images and are located at

lion dataset

	Without CC	CC + eRTC
# Chains of length 2	15,989,224	(706,364)
# Chains of length 3	0	590,325
# Chains of length 4	0	561,709
# Chains of length 5	0	593,856
# Chains of length 6	0	533,260
# Chains of length 7	0	488,775
# Chains of length 8	0	362,480
# Chains of length 9	0	163,219
# Chains of length 10	0	79,074
# Chains of length 11	0	25,944
# Chains of length 12	0	6,757
# Chains of length >12	0	675

Table 1: Evaluation of the chain length with Correspondence Chaining (CC) for the *lion dataset* with 27 images. The chains of length two are kept out.

the flank of the lions head, which is visible in a wide set of images. The majority of chains is based on five to seven images, whereas only a few 3D points rely only on three images. These last mentioned points are all close to a brink, whereas occlusions limit the number of cameras, which see these points. Chains, which are based on only two images, are kept out, because they tend to be unreliable. The chains of a given length for the *civetta dataset* and the *temple dataset* (see Figure 6) are distributed in a similar manner as for the *lion dataset*. The *temple dataset* has by trend longer chains, since it has almost the double number of input images compared to both other datasets.

Without applying the proposed Correspondence Chaining is each 3D point based on only two images, resulting in chains, which have exclusively a length of two. The *lion dataset* containes therefore without Correspondence Chaining 15,989,224 points in the pointcloud (see Table 2). With the proposed algorithm of Correspondence Chaining the number of points was reduced by 79% to 3,406,074 points, reducing directly the storage consumption. The pointcloud without Correspondence Chaining needed 1,163 MB, while the new method needs 322 MB, which is a reduction of 72%. The application of Correspondence Chaining requires additional processing time (see Table 3) for the chaining, an increase of 120% from originally 1m 14s to 2m 43s is obtained to set up all matches for triangulation. However, this calculation time is saved during the triangulation step, because due to Correspondence Chaining much less points must be triangulated: The execution time for triangulation is reduced by 67% from 8m 10s to 2m 43s. In total, the execution time for Correspondence Chaining and triangulation is reduced by 42%, while running these experiments for the *lion dataset* on an Intel Xeon W3565 with 4 cores and 3.2GHz. In summary the new method of Correspondence Chaining produces fewer 3D points by reducing redundancies of the 3D reconstruction. It outperforms the initial method in terms of higher storage efficiency and faster execution time. In Figure 7 (c) the resulting pointcloud of the *lion dataset* with Correspondence Chaining is depicted. Nearly all imprecisely reconstructed points are removed in this 3D model, as indicated by the groundtruth comparison in Figures 7(b) and 7(d).

lion dataset

	Without CC	CC + eRTC	Deviation
3D Points	15,989,224	3,406,074	- 79%
Filesize	1,163 MB	322 MB	-72%

civetta dataset

	Without CC	CC + eRTC	Deviation
3D Points	17,782,646	3,309,250	- 81%
Filesize	1,361 MB	295 MB	-78%

temple dataset

	Without CC	CC + eRTC	Deviation
3D Points	2,106,557	394,884	- 81%
Filesize	72.7 MB	32.9 MB	-55%

Table 2: Evaluation of number of points and file size for all datasets with Correspondence Chaining (CC).

lion dataset

	Without CC	With CC	Deviation
Chaining	1m 14s	2m 43s	+120%
Triangulation	8m 10s	2m 43s	-67%
Both	9m 24s	5m 26s	-42%

Table 3: Evaluation of the execution time of Correspondence Chaining (CC) for the *lion dataset*.

Overall a reduction of 3D points by 79% was performed, while the surface is still dense in most parts of the dataset. Small holes within the surface (Figures 7(c)), indicating missing 3D data, are exclusively limited to the dark parts of the input images, which are mainly reasoned by the locally concave character of the object: This does not allow for good illumination and simultaneously excludes the generation of long chains since those areas are only visible for a few cameras. Finally is the dense estimation not very reliable, since the image areas do not contain a characteristic texture for a unique matching. Therefore many pixels in this region are filtered when applying the validity check. Since the Correspondence Chaining approach leaves out chains of length two, especially points in dark areas are affected by this regulation. In Figure 8c and 9c the resulting pointclouds of the *civetta dataset* and the *temple dataset* with Correspondence Chaining are depicted. They show similar properties as the *lion dataset*. Wrongly reconstructed 3D points are removed especially around the head of the civetta and between the pillars of the temple.

3D reconstruction quality: In Figures 7(a) and 7c show the resulting pointclouds of the *lion dataset* with and without the proposed Correspondence Chaining approach, indicating the enhanced reconstruction quality. Without Correspondence Chaining a lot of 3D points are imprecisely reconstructed, but with Correspondence

lion dataset

	Without CC	With CC	Deviation
Mean Error	0.7018 mm	0.5288 mm	-25%
RMS Error	1.1752 mm	0.7461 mm	-37%

civetta dataset

	Without CC	With CC	Deviation
Mean Error	2.5620 mm	1.5470 mm	-40%
RMS Error	4.8428 mm	2.1512 mm	-56%

Table 4: Comparison of the reconstructed pointclouds of the *lion/civetta dataset* against the ground truth reconstructions of the Orcam [Köh13] and the laser scanner [Nex] respectively.

Chaining nearly all 3D points are located on the objects surface. Especially at the edges of the lion without Correspondence Chaining a lot of 3D points are wrongly reconstructed in front of the surface leading to the unsharp edges. With Correspondence Chaining nearly no flying 3D points are visible and the edges are sharp. This comparison can be found in Figure 8 and 9 for the *civetta dataset* and the *temple dataset*. Again both datasets show similar properties like the *lion dataset*. Thus, from a visual point of view the pointcloud with Correspondence Chaining is much more precisely reconstructed. To verify this enhanced precision a comparison against ground truth data is performed. For the comparison of the pointclouds against the ground truth data the one-sided Hausdorff Distance [Ruc96] was used, which is defined as

$$\sup_{x \in X} \inf_{y \in Y} d(x,y). \tag{2}$$

X represents the reference model (generated pointcloud), Y the target model (ground truth), while $d(x,y)$ holding the distance between 3D points x and y. The one-sided Hausdorff distance finds for each 3D point in the generated pointcloud the closest point in the ground truth model. Since image based 3D reconstructions are in general only up to scale, an absolute distance measure cannot be directly estimated. However, the size of the reconstruction can be mapped to a meter-scale by measuring corresponding distances in the reconstruction and on the real object.

In Table 4 the resulting pointclouds of the *lion dataset* and the *civetta dataset* with and without Correspondence Chaining are compared against ground truth data. The ground truth data for the *lion dataset* is generated by the Orcam [Köh13], which is a very accurate 3D reconstruction tool with sub-millimeter precision, while the ground truth data for the *civetta dataset* is generated by a laser scanner (NextEngine 3D Scanner HD [Nex]). The bounding box diagonal of the Lion is around 40cm and of the Civetta 50cm. In Figure 7(b) and Figure 8(b) the pointclouds without Correspondence Chaining of the *lion dataset* and the *civetta dataset* respectively are compared against the groundtruth data. Note the different scales of the two Figures. All wrongly reconstructed 3D points in front of the surface are coloured red, while correct reconstructions are shown in green. The mean error for the *lion dataset* sums up to $\sim$0.7mm, while for the *civetta dataset* a mean error of $\sim$2.6mm is acived. For the *lion dataset* we measured a root mean square error of $\sim$1.2mm and for the *civetta dataset* of $\sim$4.8mm. In Figure 7(d) and Figure 8(d) the pointclouds with Correspondence Chaining of the *lion dataset* and the *civetta dataset* respectively are compared against the groundtruth data. Much less 3D points are colored red in these Figures, i.e. 3D points with a big distance to the ground truth reconstruction are removed. The main part of the surface is colored green and fits thus to the ground truth. Only a few 3D points, which are located in holes or depressions, are colored red, because they can not be reconstructed precisely. The mean error of the *lion dataset* is reduced with Correspondence Chaining to $\sim$0.5mm, which is a reduction of 25%, while the root mean square error is reduced by 37% to $\sim$0.7mm. The mean error of the *civetta dataset* is even reduced by 40% and the root mean square error by 56%. This high reduction of both root mean square errors is an indicate that especially the points with big distance to the ground truth are reconstructed with Correspondence Chaining more precisely.

The *temple dataset* is taken from the middlebury datasets (*TempleRing*) [Sei06] and ground truth data for a self-made evaluation is not publicly available. However, from a visual point of view the precision was enhanced in a similar manner as in both other datasets.

Summarized, Correspondence Chaining reduces the redundancy of the reconstructed 3D model and the reconstructed 3D model is in average much more precise. Especially the 3D points with huge distances to the ground truth models are removed. In a next step meshes are created from the reconstructed pointclouds to verify that details are still preserved in the 3D reconstruction. Details in this context are fine structures in the surface of the object that is reconstructed.

In Figure 9 the meshes of the reconstructed pointclouds of the *temple dataset* without (9b) and with (9d) Correspondence Chaining are depicted. The meshes were created in an external tool, called MeshLab [Cig08], by using Poisson meshing (for more details [Kaz06]). Without Correspondence Chaining the surface is very rough. The stairs are almost flat, the pillars have a rough surface and the roof contains nearly no details. This is due to many imprecisely reconstructed pixels in the pointcloud, which are flying in front of the surface and which are considered by Poisson meshing since this approach is very outlier sensitive. With Correspondence Chaining (see Figure 9d) the surface is much smoother. This is due to the removed flying pixels, but the details are preserved in the mesh. The

stairs are clearly visible, the pillars contain also fine structures and the roof is full of details. We also created meshes for the *lion dataset* and the *civetta dataset*, but because of the high number of points in the pointcloud without Correspondence Chaining, around 48GB main memories were needed for meshing. With Correspondence Chaining around 10GB main memory were needed for this two datasets only. The results were similar to the *temple dataset*.

5 CONCLUSION

The introduced Correspondence Chaining approach extends existing two-view reconstruction algorithms to allow for multi-view reconstruction by chaining pairwise correspondences between images to long chains of correspondences. The correctness of the correspondences is validated using the extended Round Trip Check (eRTC), which was introduced in this work. The triangulation of long chains of correspondences is based on a wide angle and exploiting information from multiple images leading to an increased reliability of the 3D points. These claims have been evaluated on three datasets: the *lion dataset*, the *civetta dataset* and the *temple dataset*, where the applied Correspondence Chaining produced a nearly outlier free and precise 3D reconstruction. In comparison to the dense two-view reconstruction, the implemented algorithm delivers a dense multi-view reconstruction with improved precision and reduced redundancies; the enhanced results are achieved with less storage consumption and faster computation time. In the comparison with ground truth data the mean error of the reconstructed pointclouds was reduced up to a factor of 40%, whereas the root mean square error was reduced by up to 56%, indicating that especially 3D points with originally large deviations from the ground truth data are reconstructed more precisely with Correspondence Chaining. When applying the Correspondence Chaining algorithm, the computation time of the involved parts within reconstruction process (Correspondence Chaining and triangulation) was reduced by up to 42%, while file size of the reconstructed 3D models was decreased by up to 78%. The proposed Correspondence Chaining algorithm is applicable with every kind of dense estimation algorithm between image-pairs and is a starting point for further processing steps of the datasets, which rely on consistent and precisely reconstructed models.

ACKNOWLEDGEMENTS

This work was carried out in the context of a research cooperation between Sony Technology Center Stuttgart (EuTEC), DFKI, University of Padova and University of Dortmund. We would especially like to thank Yalcin Incesu and Thimo Emmerich from Sony, Matthias Brüggemann from University of Dortmund, Pietro Zanuttigh for the ground truth in the civetta dataset and Prof. Guido M. Cortelazzo for the possibility to use his private Gino Cortelazzo [Cor] collection.

6 REFERENCES

[Cig08] Cignoni, P., Callieri, M., Corsini, M., Dellepiane, M., Ganovelli, F., and Ranzuglia, G. Meshlab: an open-source mesh processing tool. In Eurographics Italian Chapter Conference, pp.129-136. The Eurographics Association, 2008.

[Cho02] Chowdhury, A. Statistical Analysis of 3D Modeling From Monocular Video Streams. PhD Thesis, University of Maryland, United States of America, 2002.

[Cor] Cortelazzo, G. [Online]. http://ginocortelazzo.it.

[Har00] Hartley, R., and Zisserman, A. Multiple view geometry in computer vision, volume 2. Cambridge Univ Press, 2000.

[Kaz06] Kazhdan, M., Bolitho, M., and Hoppe, H. Poisson surface reconstruction. In Proceedings of Eurographics symposium on Geometry processing, 2006.

[Köh13] Köhler, J., Nöll, T., Reis, G., and Stricker, D. A full-spherical device for simultaneous geometry and reflectance acquisition. In Applications of Computer Vision, pp.355-362. IEEE, 2013.

[Koc98] Koch, R., Pollefeys, M., and Van Gool, L. Multi viewpoint stereo from uncalibrated video sequences. In European Conference on Computer Vision (ECCV), 1998.

[Mer07] Merrell, P., Akbarzadeh, A., Wang, L., Mordohai, P., Frahm, J. M., Yang, R., Nister, D., and Pollefeys, M. Real-time visibility-based fusion of depth maps. In Computer Vision. IEEE, 2007.

[Mou12] Moulon, P., Monasse, P. Unordered feature tracking made fast and easy. In European Conference on Visual Media Production, 2012.

[Nex] NextEngine. 3D scanner HD. www.nextengine.com.

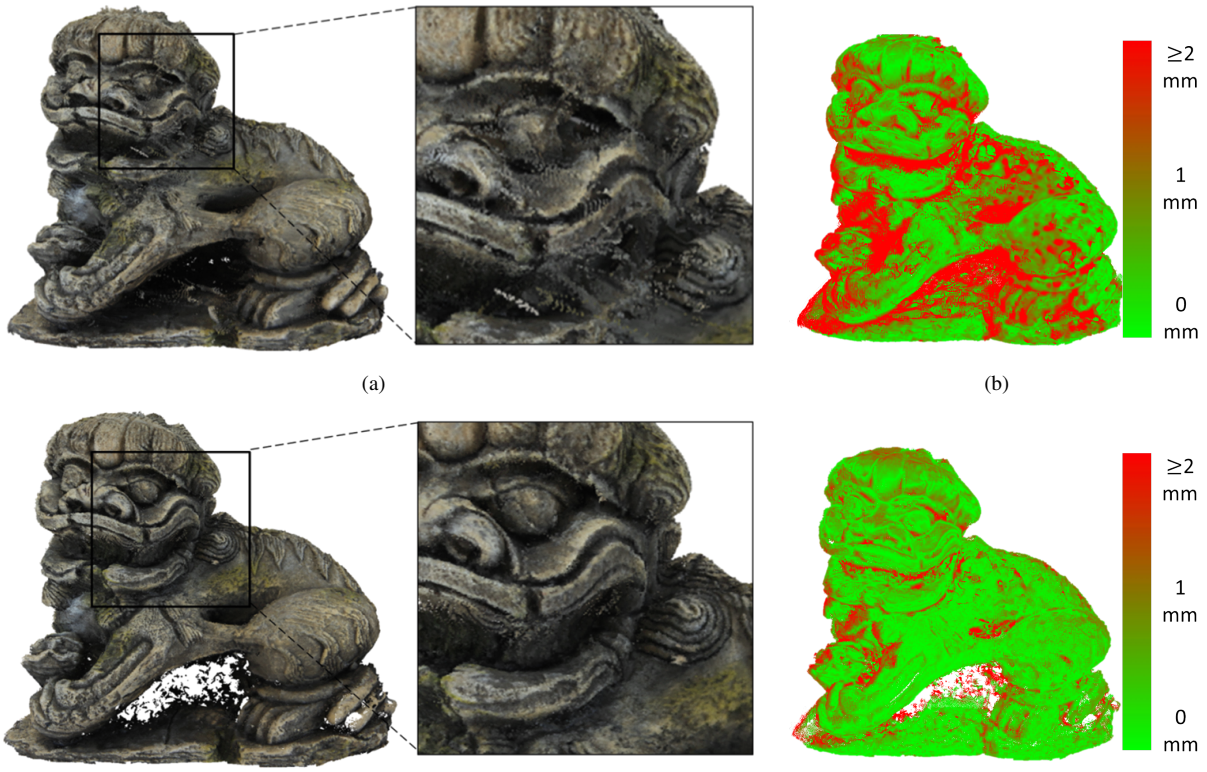
[Nöl12] Nöll, T., Köhler, J., Reis, G., and Stricker, D. High quality and memory efficient representation for image based 3D reconstructions. In Digital Image Computing Techniques and Applications, pp.1-8. IEEE, 2012.

[Rum11] Rumpler, M., Irschara, A., and Bischof, H. Multi-view stereo: Redundancy benefits for 3D reconstruction. In Workshop of the Austrian Association for Pattern Recognition, 2011.

[Ruc96] Rucklidge, W. Efficient visual recognition using the Hausdorff distance. Springer Heidelberg, 1996.

[Sei06] Seitz, S., Curless, B., Diebel, J., Scharstein, D., and Szeliski, R. A comparison and evaluation of multi-view stereo reconstruction algorithms. In Computer vision and pattern recognition, pp.519-528. IEEE, 2006.

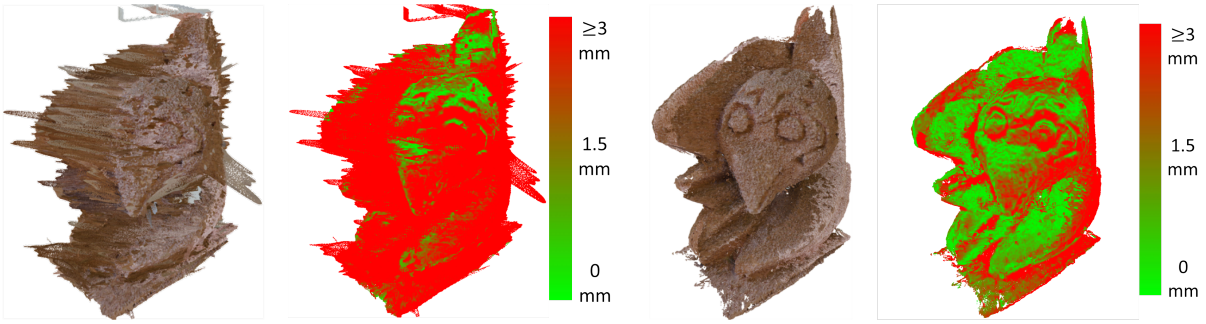
[Sun03] Sun, J., Zheng, N. N., and Shum, H. Y. Stereo matching using belief propagation. In Pattern Analysis and Machine Intelligence. IEEE, 2003.



(a) (b)

(c) (d)

Figure 7: Reconstruction results for the *lion dataset* (27 images) accompanied by color-encoded comparisons to the corresponding groundtruth: Without (a,b) and with (c,d) the proposed Correspondence Chaining algorithm.



(a) (b) (c) (d)

Figure 8: Reconstruction results for the *civetta dataset* (28 images) accompanied by color-encoded comparisons to the corresponding groundtruth: Without (a,b) and with (c,d) the proposed Correspondence Chaining algorithm.

(a) (b) (c) (d)

Figure 9: Reconstruction results for the *temple dataset* (47 images) accompanied by visualizations of polygon meshes created on the basis of the pointclouds: Without (a,b) and with (c,d) the proposed Correspondence Chaining algorithm.