

IMPROVING 3D RECONSTRUCTION TRACKS USING DENOISED EUCLIDEAN DISTANCE MATRICES

S. Milani

University of Padova,
Dept. of Information Engineering,
e-mail: simone.milani@dei.unipd.it

ABSTRACT

The reconstruction of 3D point cloud models from unordered and uncalibrated sets of images has recently been a hot topic in the computer vision world. Most of the proposed solutions rely on the Structure-From-Motion algorithms, and their performances are significantly affected by the processing order (called *track*) of the considered images. This is computed according to a distance (or similarity) metric between couples of images, which is usually highly noisy.

The paper proposes an image ordering strategy that models the distances between images as an Euclidean distance matrix and applies a rank-based denoising algorithm in order to refine the metric values. Experimental results prove that the accuracy of the final 3D model is sensibly improved.

Index Terms— Structure-from-Motion, denoising, Euclidean distance matrix, descriptor matching, 3D reconstruction

1. INTRODUCTION

Over the Internet, it is possible to find billions of images taken in the same place by different users with different light conditions, capture configurations, daytime, etc. During the last years the computer vision community has been investigating several strategies that process this massive amounts of data to infer a deeper understanding about the location and its characteristics [1, 2].

One of the major goals of such approaches is the three-dimensional reconstruction of real environments generalizing the Structure-from-Motion (SfM) strategies to heterogeneous set of unordered and uncalibrated images [3–5]

As a matter of fact, 3D reconstruction and processing strategies needs to face the problem of image ordering, i.e., how to progressively-include the images in the algorithm. This order is called *track* [3] and significantly affects the quality of reconstruction. Figure 1 reports the 3D point cloud models estimated by the algorithm in [3] with different tracks. It is possible to notice moving from one image to the



Fig. 1. Point clouds generated using different tracks. a) progressively including the geometrically-closest image; b) random ordering.

geometrically-closer one in the set permits having a finer reconstructed model, i.e., the right model presents many holes, flying pixels, and artifacts. Most of the proposed solutions compute the ordering from a correspondence (or distance) matrix, i.e., a set of measurements parameterizing the similarity between couples of images and organized in a matrix structure [6]. Unfortunately, these data result to be noisy and poorly reliable [7].

The paper presents a matrix denoising strategy which is based on modeling the image correspondence matrix as an Euclidean distance matrix. The resulting image order permits generating a more accurate and denser point cloud with a negligible additional computational cost. The remaining of the paper is organized as follows. Section 2 reviews some of the state-of-the-art techniques presented literature and highlights the main problems. Section 3 shows how it is possible to model the image correspondence matrix as an Euclidean distance matrix, and Section 4 presents the denoising strategy. Section 5 reports the performance of the proposed solution in terms of accuracy, and Section 6 draws the final conclusions.

2. 3D MODELIZATION VIA STRUCTURE-FROM-MOTION: RELATED WORKS AND MAIN ISSUES

One of the first strategies to be presented is the *Bundler* algorithm [3], which starts estimating a 3D point cloud model of the scene by coupling pairs of images according to the matching local features [8]. Similar to [3], the VisualSfM

The work has been supported by the Robotic 3D and by the 3D Cloud-Vision projects, funded by the University of Padova, Italy.

software [5] implements a multicore bundle adjustment using GPU-optimized SIFT.

For every image $I_i \in \mathcal{I}$ ($i = 0, \dots, N-1$), the approach in [3] compute a set of n_i SIFT keypoints $S_i = \{m_{i,k}, k = 0, \dots, n_i\}$, where the pixel position $\mathbf{m}_{i,k}$ is associated to the k -th descriptor $\mathbf{s}_{i,k}$. Moreover, it is possible to assume that, without loss of generality, the pixel $\mathbf{m}_{i,k}$ is also associated to the a 3D point \mathbf{P}_k via a pinhole camera model [9].

For every couple of images I_i, I_j , it is possible to generate the set $S_{i,j} = \{(\mathbf{m}_{i,k}, \mathbf{m}_{j,h}) \mid \mathbf{s}_{i,k} \text{ matches } \mathbf{s}_{j,h}\}$. If no wrong matches are present, $k = h$ for all the couples in $S_{i,j}$ (i.e., they are associated to the same 3D point).

At this point, both camera parameters and a sparse 3D model can be reconstructed via a resection-intersection strategy [10] and then refined by a bundle adjustment strategy [11, 12] implemented via a least-square minimization.

This initial 3D model can then be refined and densified including in the reconstruction process additional images following a specific track. This sequence can be crucial for the quality of the final point cloud since the first links that are included in the model have a stronger impact on the 3D point estimation [13]. Therefore, it is necessary to find an ordering that avoids processing inconsistent or weak links before strong ones.

Image ordering is defined by a metric that parameterizes the similarity between couples of images. A weighted connectivity graph is then created (represented by a correspondence or a distance matrix), and according to edge label values that are associated to image difference, an optimal sequence is inferred.

The approach in [3] estimates the fundamental matrix using the 8-points algorithm; this estimation permits removing outliers and generating the new set of matched points $S'_{i,j}$. The final correspondence matrix is then $C = [c_{i,j}]_{i,j}$, where $c_{i,j} = |S'_{i,j}|$, and the track is generated by ordering $c_{i,j}$ in decreasing order.

Differently from Bundler, the solution in [4] first clusters images in a binary agglomerative tree using a modified hierarchical clustering.

A later improvement of the mentioned approach [14] divide the image clustering in two steps: a broad phase (using a reduced set of matched SIFT keypoints) and a narrow phase (which refines the results of the previous step).

All these solutions imply refining the distance (or correspondence) estimate for every couple of images using complex algorithms [15], like RANSAC and MSAC. These solutions work well whenever the number of correctly-matched keypoints is very high with respect to outliers. Whenever this condition is not verified, the estimated distances are not reliable.

The main innovation of the proposed approach stands on the fact that it is possible to achieve more accurate distance estimates by applying a denoising strategy on the image distance matrix itself with a much lower computational complex-

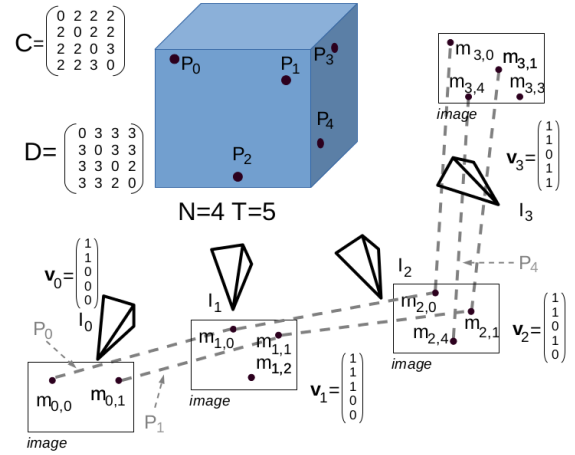


Fig. 2. Example of correspondence matrix C in a multiview setting with $N = 4$ and $T = 5$.

ity. The ordering derived from the denoised data leads to more accurate and denser models as Section 5 is going to show. In order to perform this, the correspondence matrix can be modelled as an Euclidean distance matrix.

3. MODELING IMAGE CORRESPONDENCES AS EUCLIDEAN DISTANCES

Given the set \mathcal{I} of N images, it is possible to generate a distance matrix $D = [d_{i,j}]_{i,j=0,\dots,N-1}$ with $d_{i,j} = T - c_{i,j}$ when $i \neq j$ and $d_{i,i} = 0$. Parameter T is the total number of 3D points associated to a SIFT keypoints in the model, viz. $T \geq c_{i,j} \forall i, j$. Assuming that the 3D points $\mathbf{P}_t, t = 0, \dots, T-1$ are indexed and ordered, for every image it is possible to generate a binary array \mathbf{v}_i of length T where the t -th element $v_{i,t}$ is 1 if exists a keypoint $m_{i,k} \in S_i$ that is projection of \mathbf{P}_t . It is straightforward to see that

$$d_{i,j} = T - c_{i,j} = \|\mathbf{v}_i - \mathbf{v}_j\|^2, \quad (1)$$

and therefore, the matrix D is an Euclidean distance matrix or EDM. Figure 2 reports a simple example with $N = 4$ and $T = 5$.

From this assumption, it can be proved that the geometric-centered version D_C of noise-free Euclidean distance matrix D must be positive semidefinite [16], i.e.,

$$D_C = -\frac{1}{2} J D J. \quad (2)$$

The matrix $J = I - 1/N \cdot \mathbf{1}^T \mathbf{1}$ is the geometric centering matrix, with I being the identity matrix and $\mathbf{1}$ the column vector of all ones.

The matrix D_C is positive semidefinite if all its eigenvalues λ_i are greater or equal to 0, i.e., $\lambda_i \in \mathbb{R}_+$. The rank of D_C

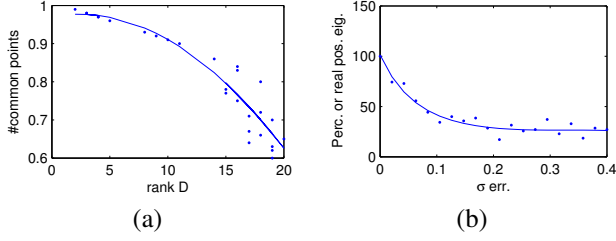


Fig. 3. Properties of image distance matrices. a) Rank of D_C vs. the number of points \mathbf{P}_t which are seen by all the cameras, i.e., $v_{i,t} = 1 \forall i$; b) Number of positive eigenvalues vs. noise variance σ .

depends on the number of 3D points \mathbf{P}_t that are projected on the image planes of all the I_i . Figure 3a reports the rank vs. the number of common points generated in a simulated multicamera scenario, where the acquiring devices are randomly placed around the scene. It is evident that a low rank matrix implies good matchings between different pictures.

Unfortunately, real D_C matrices are not always positive semidefinite since image distance metric presents significant levels of noise. This is due to the false keypoint matches, which are not completely filtered out by MSAC or the RANSAC algorithm. Figure 3b reports the percentage of positive λ_i vs. the noise variance σ , where Gaussian zero-mean noise was added to point coordinates of the previous simulated multicamera scenario in order to reproduce more realistic conditions. The graph shows that noisy measurements significantly impair the positive semi-definiteness of D_C , but it is possible to recover this property by processing the matrix D with rank-based denoising algorithms.

Some of these strategies have been introduced for sensor network applications, but in this case, they are to be modified to preserve distance values with limited noise and employ in the 3D reconstruction the connected points that are present in most images. To this purpose, the alternating rank algorithm [16] was modified as described in the following section.

4. THE DENOISING ALGORITHM





Given the initial distance matrix D , it is possible to recover a more accurate version by removing the noisy elements and filling them using a rank-completion strategy. The distance values of D can be selected by introducing the matrix $W = [w_{i,j}]$, where $w_{i,1} = 1$ if the distance $d_{i,j}$ is to be used or $w_{i,1} = 0$ if $d_{i,j}$ is to be removed. This allows generating the matrix

$$D_W = W \circ D + (\mathbf{1}^T \mathbf{1} - W)\mu, \quad (3)$$

where \circ denotes the element-by-element multiplication.

All the distances to be removed are replaced by the initialization constant μ . Then, matrix D_W is decomposed into $U L U^T$, where L is the diagonal matrix of eigenvalues λ_i (Eigen Valued Decomposition or EVD). Let us assume that

Table 1. Datasets adopted in the experimental tests.

Dataset	Name	Description
	ball	29 pictures; reference point cloud model acquired with a laser scan; indoor.
	portello	40 pictures; reference point cloud model generated from 80 images; outdoor.
	notredame	35 pictures; reference dataset downloaded from [18]; outdoor.
	tiso_palace	30 pictures; reference point cloud model generated from 100 images; outdoor.

the values λ_i are ordered in non-increasing order, i.e., $\lambda_i \geq \lambda_{i+1}$. The algorithm changes the matrix L into L' by keeping only the d most significant eigenvalues in L and setting the others to zero. Then, the matrix $D'_W = U L' U^T$ is computed, and the procedure is iterated until convergence.

These operations are summarized by the following pseudo-code:

```

function rankCompletion( $D_W, d$ )
  repeat
     $D_{W0} \leftarrow D_W$ ;
     $U, L = \text{EVD}(D_{W0})$ ;            $\# L = \text{diag}(\lambda_1, \dots, \lambda_N)$ 
     $L' \leftarrow \text{diag}(\lambda_1, \dots, \lambda_d, 0, \dots, 0)$ 
     $D_W \rightarrow U L' U^T$ 
     $D_W \leftarrow W \circ D_{W0} + (\mathbf{1}^T \mathbf{1} - W) \circ D_W$ 
     $\text{diag}(D_W) \leftarrow 0$             $\# \text{set diagonal to 0}$ 
     $D_{W-} \leftarrow 0$               $\# \text{set negative dist. to 0}$ 
  until  $\|D_W - D_{W0}\|_F < \epsilon$ 
  return  $D_W$ 

```

Note that the alternating rank-based EDM algorithm [16] implies the knowledge of the desired rank d . For a generic distance matrix D , this information is not available, but low rank configurations are desirable since they correspond to keypoints matching in multiple images. To this purpose, it is introduced the indexing function $u_d(t) : \mathbb{Z} \mapsto \mathbb{Z}^2$ that maps indexes t to couples (i, j) s.t. $t < h$ implies $d_{u_d(t)} < d_{u_d(h)}$.

Then, the complete denoising algorithm is:

```

 $\ell \rightarrow \ell_{\min}$ 
while  $-\frac{1}{2} J D J$  is not pos. semidef.  $\wedge \ell \leq N^2$  do
  take the first  $\ell$  values
  generate  $W$  such that  $w_{u_d(t)}$  is 1 for  $t = 0, \dots, \ell - 1$  and 0 otherwise
   $D_W = W \circ D + (\mathbf{1}^T \mathbf{1} - W)\mu$ 
   $d = \text{rank}(D_W)$ 
  rankCompletion( $D_W, d$ )
   $D \leftarrow D_W$ 
   $\ell \leftarrow \ell + 1$ 
end while

```

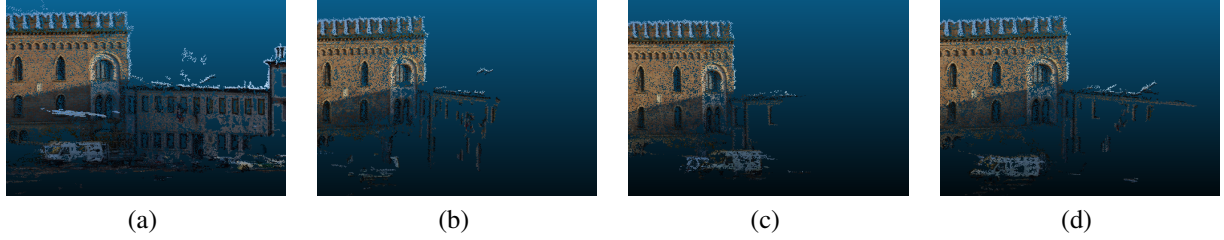


Fig. 4. Details of point cloud models computed on `tiso_palace` dataset using different algorithms. a) Reference model; b) Bundler; c) Hierar. clustering; d) Hierar. clust. with denoise.

After denoising D , it is possible to build the track. In this approach, distances $d_{i,j}$ are increasingly ordered and progressive agglomerative clusters of images are created minimizing the mean distance between the clusters (similarly to [4]). Starting from the initial cluster made of the couple of pictures with minimum distance, it is possible to generate a track by including images in the sequence as they are merged with this initial cluster. The effectiveness of this strategy is evaluated in the following section.

5. EXPERIMENTAL RESULTS

In order to evaluate the performances, the proposed approach was tested on different image datasets summarized in Table 1. Dataset `ball` was acquired in an indoor controlled set-up, while the others are referred to outdoor acquisitions. Each image set is associated to a ground truth point cloud model, which was generated from larger set of images taken with high quality cameras. As for dataset `ball`, the ground truth point cloud was acquired using NextEngine Desktop Laser Scan. Datasets are available at [17].

The reconstructed three-dimensional point clouds were generated using the Bundler software [3], which was modified in order to process different tracks. Each point cloud is then resampled using the PMVS2 software [19] in order to generate denser point clouds. The generated 3D models were compared to the reference point clouds using the CloudCompare software [20]. At first, the software is used to estimate an affine transformation between sets of matched points. Then, this transformation is refined using the ICP algorithm [21]. Then, each point in the model was associated to a point in the reference model using the KNN algorithm [22]. Table 5 reports mean and the variance of distances between reference model and the point clouds computed by different algorithms. Experimental data show that hierarchical clustering permits improving the quality of the reconstructed 3D model up to one order of magnitude (see the average distance for datasets `notredame` and `tiso_palace`). Using a denoised matrix permits improving the accuracy of the point cloud for all the dataset. Figure 4 reports a detail from the models reconstructed by different algorithms for the `tiso_palace` dataset. It is possible to notice that the 3D points in Fig. 4d

Table 2. Mean and variance of distances between reference point clouds and estimated models ($\times 10^{-4}$).

Dataset	Bundler [3]	Hierar. clustering	Hierar. clust. with denoise
<code>ball</code>	0.60 ± 0.27	0.62 ± 0.28	0.05 ± 0.03
<code>tiso_palace</code>	2.75 ± 1.58	0.81 ± 0.45	0.04 ± 0.02
<code>notredame</code>	9.86 ± 5.37	1.07 ± 0.59	0.78 ± 0.43
<code>portello</code>	3.57 ± 1.70	0.56 ± 0.41	0.19 ± 0.11

are much closer to the reference model in Fig. 4a. The models generated by the Bundler (Fig. 4b) and by adding images using hierarchical clustering (Fig. 4c) are sparser, and, according to the data in Table 5, less accurate. Further results and the adopted datasets are available at [23].

Finally, the reconstruction accuracy using MSAC-based distances [14] between images was evaluated on the `ball` and `tiso_palace` dataset. The final average distances were 0.03 ± 0.01 and 0.05 ± 0.03 ($\times 10^{-4}$), which make this performance comparable with that of the proposed approach. Unfortunately, computational complexity for MSAC algorithms was 9,782 times higher with respect to the proposed solution for the `tiso_palace` dataset and 8,450 times higher for the `ball` dataset. This makes the MSAC approach prohibitive for large scale datasets.

6. CONCLUSIONS

The paper presents a denoising strategy for image distance matrix in reconstructing three-dimensional environments from unordered image collections. The solution re-estimates noisy distance measurements by iteratively imposing positive semidefiniteness properties on the geometric-centered distance matrix. Experimental results show that the generated point cloud models are more accurate, dense, and presents a lower amount of flying pixels. Since the denoising solution can be applied to a generic distance matrix, future works will be devoted to test its performance on different distance metrics.

7. REFERENCES

- [1] M. Valt, R. Salvatori, P. Plini, R. Salzano, M. Giusto, and M. Montagnoli, "Climate change: A new software to study the variations of snow images shot by web cam," in *Proc. of ISSW 2013*, oct 2013.
- [2] Simone Milani, "Compression of multiple user photo galleries," *Image and Vision Computing, Special issue on Event-based Media Processing and Analysis*, vol. 53, pp. 68 – 75, Sept. 2016, Available at: <http://www.sciencedirect.com/science/article/pii/S0262885615001407> [Online].
- [3] Noah Snavely, Steven M. Seitz, and Richard Szeliski, "Modeling the world from internet photo collections," *Int. J. Comput. Vision*, vol. 80, no. 2, pp. 189–210, Nov. 2008.
- [4] R. Gherardi, M. Farenzena, and A. Fusiello, "Improving the efficiency of hierarchical structure-and-motion," in *Proc. of CVPR 2010*, June 2010, pp. 1594–1600.
- [5] ChangChang Wu, "VisualSFM : A Visual Structure from Motion System," <http://ccwu.me/vsfm/>, Dec. 2014.
- [6] Bill Triggs, "Joint feature distributions for image correspondence," in *Proc. of ICCV 2001*, Vancouver, Canada, July 2001, vol. 2, pp. 201–208, IEEE.
- [7] A. Melloni, P. Bestagini, S. Milani, M. Tagliasacchi, A. Rocha, and S. Tubaro, "Image phylogeny through dissimilarity metrics fusion," in *Proc. of EUVIP 2014*, Dec 2014, pp. 1–6.
- [8] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
- [9] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, ISBN: 0521540518, second edition, 2004.
- [10] R. Lakemond, C. Fookes, and S. Sridharan, "Resection-intersection bundle adjustment revisited," *ISRN Machine Vision*, vol. 2013, no. Article ID 261956, Apr. 2013.
- [11] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment a modern synthesis," in *Vision Algorithms: Theory and Practice*, vol. 1883 of *LNCS*, pp. 298–372. Springer Berlin Heidelberg, 2000.
- [12] S. Milani, "Three-dimensional reconstruction from heterogeneous video devices with camera-in-view information," in *Proc. of ICIP 2015*, Sept. 2015, pp. 2050–2054.
- [13] S. Gammeter, T. Quack, D. Tingdahl, and L. J. Van Gool, "Size does matter: Improving object recognition and 3d reconstruction with cross-media analysis of image clusters," in *Proc. of ECCV 2010*. 2010, vol. 6311 of *Lecture Notes in Computer Science*, pp. 734–747, Springer.
- [14] R. Toldo, R. Gherardi, M. Farenzena, and A. Fusiello, "Hierarchical structure-and-motion recovery from uncalibrated images," *Comput. Vis. Image Underst.*, vol. 140, no. C, pp. 127–143, Nov. 2015.
- [15] R. Szeliski, "Image alignment and stitching: A tutorial," *Found. Trends. Comput. Graph. Vis.*, vol. 2, no. 1, pp. 1–104, Jan. 2006.
- [16] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean distance matrices: Essential theory, algorithms, and applications," *Signal Processing Magazine, IEEE*, vol. 32, no. 6, pp. 12–30, Nov 2015.
- [17] S. Milani, "Photo dataset for 3D reconstruction - <http://www.dei.unipd.it/~sim1mil/materiale/photoset/>, 2014.
- [18] Cornell University, National Science Foundation, Amazon Web Services in Education, MIT Lincoln Labs, Microsoft, and IU Data to Insight Center, "BigSFM: Reconstructing the World from Internet Photos," <http://www.cs.cornell.edu/projects/bigsfm/#data>, 2015, [Online].
- [19] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1362–1376, 2010.
- [20] EDF R& D, "Cloud compare," <http://www.cloudcompare.org/>, 2015, [Online].
- [21] F. Pomerleau, F. Colas, and R. Siegwart, "A review of point cloud registration algorithms for mobile robotics," *Found. Trends Robot*, vol. 4, no. 1, pp. 1–104, May 2015.
- [22] G. Shakhnarovich, T. Darrell, and P. Indyk, *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice (Neural Information Processing)*, The MIT Press, 2006.
- [23] S. Milani, "3D Reconstruction Datasets," <http://www.dei.unipd.it/~sim1mil/materiale/3Drecon>, 2015, [Online].