Semi-Global 3D Line Modeling for Incremental Structure-from-Motion

Manuel Hofer
hofer@icg.tugraz.at
Michael Donoser
donoser@icg.tugraz.at
Horst Bischof
bischof@icg.tugraz.at

Institute for Computer Graphics and Vision
Graz University of Technology
Austria

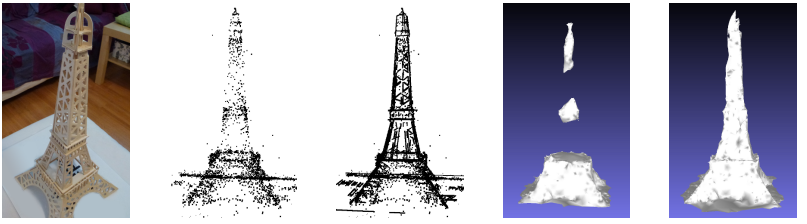
Abstract

Structure-from-Motion (SfM) approaches, which are conventionally based on local interest point matches, tend to work well for richly textured indoor- and outdoor environments. However, in less textured scene areas the density of the resulting point cloud suffers from the lower number of matchable interest points. This significantly affects subsequent computer vision tasks like image based localization, surface extraction or visual navigation. In this paper, we propose a novel 3D reconstruction approach that increases the amount of 3D information in the reconstruction by exploiting line segments as complementary features. We introduce an efficient and effective semi-global approach, which takes into account local (per 2D line segment) as well as global (graph clustering) 3D line hypotheses constellations. Our approach outperforms the state-of-the-art in terms of accuracy, with comparable runtime.

1 Introduction

Recovering 3D information from a single moving camera is a widely studied field in the area of computer vision [1, 9, 21, 23, 24]. Most of these Structure-from-Motion (SfM) approaches are based on so-called interest points (e.g. corners) in images, which can be accurately matched using powerful descriptors like SIFT [18]. Hence the output is usually a sparse 3D point cloud along with the camera poses for all successfully integrated images. While previous methods were only able to perform pose estimation and 3D reconstruction in an offline way, there are now more and more incremental SfM approaches available [13, 20, 27, 28].

Since conventional SfM approaches are based on interest points, the distribution of the obtained 3D points is usually not uniform throughout the whole reconstruction. This is due to the fact that such interest points are usually located on highly textured areas, but not on homogeneous regions or along edges. Since the result of SfM pipelines is often used as basis to generate a more dense result or for localization and navigation tasks, it would be beneficial to generate additional complementary 3D information in an efficient way. From a SfM point of view, using line segments is especially interesting for urban and indoor environments, where linear structures frequently occur. While interest points are located mostly on richly textured image locations, line segments usually mark the boundaries of objects. Hence,



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

Figure 1: (a) An example image from the *EIFFEL* sequence. (b) The sparse 3D reconstruction result obtained by a conventional point-based SfM pipeline [13]. (c) The pointcloud combined with reconstructed 3D lines by our proposed method. On the right we can see an incrementally generated 3D mesh with (d) the 3D points only or (e) both points and lines. As we can see, the usage of complementary features significantly improves the completeness of the resulting 3D model in both cases.

incorporating such features in an online SfM pipeline to create 3D line segments naturally leads to a more complete 3D representation of the underlying scene, which is beneficial for all kinds of subsequent applications.

We propose a novel approach which generates 3D line models on-the-fly, based solely on the output of a conventional incremental SfM pipeline. The goal of our method is to generate additional complementary 3D information to improve the sparse 3D representation of the scene. In this approach, we consider the SfM pipeline as a black box and do not interfere with the pose estimation procedure. We show that 3D line reconstructions can be obtained very efficiently by using purely geometric constraints, or by additionally incorporating appearance and collinearity information. Our approach enables accurate 3D reconstruction of texture-less as well as textured man-made objects, including complex structures such as wiry objects. Figure 1 shows a reconstruction result obtained by an incremental SfM system [13], followed by a surface generation method [14], with and without the usage of additional 3D line segments obtained by our proposed method. As we can see, additional 3D information significantly improves the completeness and overall appearance of the resulting reconstructions.

Conventional line-based 3D reconstruction methods usually require some sort of explicit one-to-one line-segment matching (e.g. by using normalized cross correlation scores [2, 15] or line descriptors [4, 5, 10, 17, 29, 30, 31]), or very specific scene structures [22]. Despite the reasonable matching scores when using line descriptors, their patch-based nature is not beneficial when segments from very complex structures have to be matched. A prominent example are wiry structures such as power pylons, cell phone towers or scaffolds. For this kind of objects, patch based line descriptors fail to achieve a reasonable performance, since we have to deal with highly viewpoint depending surroundings. While such structures usually contain a low amount of distinctive interest points as well, it is usually possible to at least compute the correct camera poses with traditional SfM methods (e.g. due to correct feature matches on the ground or behind the wiry object). This is also exploited by the current state-of-the-art in incremental line-based 3D reconstruction [12], which builds up on appearance-less 3D reconstruction methods based on given camera poses [11, 16]. In [12] an incremental SfM is combined with a purely geometric 3D line segment clustering approach. The core principle is to compute a potentially large set of 2D line segment matches among neighboring images using weak epipolar constraints, and to compute 3D line segment hy-

potheses for all of these matches. In contrast to related offline approaches [11, 16], where only the best hypothesis for each 2D line segment was kept for the final clustering, in [12] all 3D hypotheses are treated as equal and a direct clustering in space is done whenever two hypotheses are closer than a certain grouping radius.

Despite the fact that the pipeline of [12] delivers visually pleasant results, there is always a trade-off between accuracy and completeness of the resulting 3D line model. On the one hand, choosing a small spatial grouping radius ensures an outlier free result, but depending on the triangulation uncertainty and the configuration of the camera poses it is often the case that several relevant parts of the object are not captured in the reconstruction. On the other hand, increasing the grouping radius quickly leads to noisy reconstructions with a significantly higher amount of gross outliers, which weakens all kinds of post-processing tasks. The problem is that direct clustering does not reflect the global constellation of the hypotheses. Hence, it might be possible that a cluster emerges at the wrong location in space just by chance.

To overcome these drawbacks we propose a novel reconstruction approach (denoted as *semi-global line modeling*), which takes into account local (per 2D line segment) as well as global hypotheses constellations (graph clustering), instead of making only greedy decisions. Furthermore, we will show how we can adapt the idea of deriving a spatial direct grouping radius from the image space to formulate affinities for potentially corresponding hypotheses. Additionally, we extend the epipolar geometry based matching procedure to include distant line segments, which might be collinear with an already matching segment. This enables us to jointly optimize non-overlapping line segments, which have emerged from the same 3D line in space (e.g. window frames). As a final contribution, we introduce a boosted version of the HSV histogram based line matching method by Bay et al. [3], and show how it can be optionally utilized to further refine the set of pairwise matches, even for wiry objects.

In Section 2 we will introduce some notation and lay the theoretical foundation, before we explain all necessary computation steps in more detail. We will conclude with extensive experimental results in Section 3.

2 Semi-Global 3D Line Modeling

Given an unordered set of images $I = \{I_1, \cdots, I_N\}$ and the corresponding camera poses $C = \{C_1, \cdots, C_N\}$ (obtained by any conventional SfM pipeline), our goal is to reconstruct an accurate and complete 3D line model. Furthermore, we define a set $\hat{I}_M(i) \subset I \backslash \{I_i\}$ for each image I_i, which contains its M nearest visual neighbors (e.g. the M images with the highest amount of common worldpoints with I_i). Additionally we assume that we have a set of 2D line segments $L_i = \{l_{i,1}, \cdots, l_{i,n_i}\}$ per image I_i, where n_i refers to the number of line segments in I_i.

Our method consists of several steps. In Section 2.1 we show how to compute matches between 2D line segments across neighboring images, to generate a potentially large set of 3D line hypotheses. In Section 2.2 we show how to select the locally best 3D hypothesis for each 2D segment, to reduce the number of hypotheses to be evaluated. In Section 2.3 we introduce the graph-based 3D hypothesis clustering procedure to merge corresponding hypotheses together. Finally, we discuss how to compute incremental reconstruction results in Section 2.4.

2.1 3D Line Hypotheses Estimation

To compute an accurate 3D line model, we first need to match potentially corresponding 2D observations (2D line segments) across neighboring image pairs. Since we cannot expect

(a) Wiry structure (b) Solid structure

Figure 2: The red segment on the left-hand side denotes an exemplar line segment from an image I_i. On the right we can see the corresponding epipolar lines (blue) along with the potential matches in I_j. The yellow segments fulfill the geometric matching conditions only, while the green segments are also similar in color. In both cases, the number of potential matches has been significantly reduced.

to compute outlier free one-to-one matches when complex objects (e.g. wiry structures) are present, we compute multiple matching hypotheses for each line segment. To ensure a high recall with a reasonable precision, we use simple epipolar matching conditions and optionally incorporate appearance information and collinearity constraints. We first compute binary matching matrices $M_{i,j}$ $(n_i \times n_j)$ for all image pairs I_i and $I_j \in \hat{I}_M(i)$ by using the same geometric matching constraints as presented in [12]. These constraints check that (a) at least one of the endpoints of $l_{j,m}$ must be close to its nearest epipolar line corresponding to the other segment, and (b) the orientation must be correct. Despite the high recall of this purely geometric matching scheme, the precision can be arbitrarily low. To improve the line matching performance, we adapt a line matching method by Bay et al. [3], recently revisited in [25]. The algorithm is based on HSV color histograms and does not require a patch-based support region. Therefore, we compute two separate HSV color histograms $\Psi^L_{i,n}$ and $\Psi^R_{i,n}$ (left and right) for each line segment $l_{i,n}$, using two stripes directly adjacent to the line segment on both sides. In contrast to the similarity metric proposed in [3], we use the faster to compute *Jensen-Shannon divergence (JSD)* as a similarity measure, defined as

$$d(\Psi_1,\Psi_2) = \frac{1}{2}\sum_x \left(log_2\left(\frac{\Psi_1(x)}{\Delta(x)}\right)\Psi_1(x) + log_2\left(\frac{\Psi_2(x)}{\Delta(x)}\right)\Psi_2(x)\right), \quad \Delta = \frac{\Psi_1+\Psi_2}{2} \tag{1}$$

where x stands for the histogram bins. To be more robust against illumination changes, we apply a certain amount of histogram smoothing. Therefore, we use the Euclidean distances between the histogram bins for a soft assignment to the actual bin and its k nearest neighbors. Since it is often the case that the color profile is only discriminative at one side of the segment, we compute two separate similarity measures for both sides. We then define the similarity $d_{sim}(l_{i,n}, l_{j,m})$ to be the minimum of these two measurements. Non-zero entries in $M_{i,j}$, which correspond to segment pairs where $d_{sim}(l_{i,n}, l_{j,m}) \geq t_{sim}$ are subsequently removed. Figure 2 shows a comparison between the purely geometric matching and an additional verification using the modified histogram matching. As we can see, the procedure significantly reduces the number of potential matches.

Due to the epipolar matching constraints, line segments that are located on the same infinite line in space but do not have a spatial overlap (e.g. aligned window frames) would not be matched together. To overcome this drawback, we incorporate potential collinearity information. We therefore create a binary collinearity map P_i $(n_i \times n_i)$ per image, which is set to one if the corresponding 2D segments have a maximum distance smaller than σ_c, and are visually similar (using their HSV histograms). Now, for each pair of segments $l_{i,n}$ and

$l_{j,m}$ that are currently not matching (i.e. $M_{i,j}(n,m) = 0$), we check if there exists a potentially collinear segment $l_{j,\bar{m}}$ which does (i.e. $M_{i,j}(n,\bar{m}) = 1$ and $P_j(m,\bar{m}) = 1$). If so, we consider $l_{i,n}$ and $l_{j,m}$ to be matching as well and update the matching matrices accordingly. This procedure enables us to match separate line segments from the same 3D line together, with minimal additional effort.

The aforementioned matching procedure gives us a potentially large number of pairwise matches. To verify which of them actually belong together (i.e. which of them are observations from the same physical 3D line structure), we transform them into the 3D space and apply spatial clustering. Given the pairwise matching matrices $M_{i,j}$ for all image pairs (I_i, I_j) with $I_j \in \hat{I}_M(i)$, we compute a 3D line segment hypothesis $h_{i,j}^{n,m}$ for each matching pair, by triangulating the corresponding 2D segments, as shown in [11, 12]. Since we also have spatially distant matches based on the collinearity estimation, we compute two separate collinear 3D line segments, $s_{i,j}^{n,m}$ and $s_{j,i}^{m,n}$. The hypothesis $h_{i,j}^{n,m}$ is then defined as $h_{i,j}^{n,m} = \left\{ s_{i,j}^{n,m}, s_{j,i}^{m,n} \right\}$.

In general, we cannot verify or discard single hypotheses at this point. Though, as stated in [12], we can estimate a quality measure θ based on the visibility of a triangulated hypothesis as

$$\theta\left(h_{i,j}^{n,m}\right) = 1 - \min_q \left\{ \left| \left(\overrightarrow{s_{i,j}^{n,m}}\right)^T \cdot \overrightarrow{c_q} \right| \right\}, \quad q \in \{i,j\} \tag{2}$$

where $\overrightarrow{s_{i,j}^{n,m}}$ and $\overrightarrow{c_q}$ are the unit directional vectors of the triangulated segments and the optical axes of the cameras C_i and C_j respectively. This formulation assigns a high quality value θ to hypotheses, that have a large angle between the 3D line and the optical axis of one of the supporting cameras. Hypotheses with a low quality θ are in general less likely to be correct because the contributing 2D line segments in the image space appear very small, in contrast to their 3D equivalent. Even though such hypotheses could also be correct, the triangulation quality is usually very poor for such cases. Therefore, it is beneficial to discard low-quality hypotheses (e.g. $\theta\left(h_{i,j}^{n,m}\right) < 0.5$) at this point. Even though this is not strictly necessary when using our semi-global method, it is strongly recommended since the runtime is decreased while the results are not negatively affected. Finally, all hypotheses that remain valid are put into the hypotheses set H. For invalidated hypotheses we set the corresponding entries in the matching matrices $M_{i,j}$ to zero. Given this information we can now proceed to the task of hypothesis verification and clustering.

2.2 Local Hypothesis Selection

Given the hypotheses set H we could directly apply a spatial clustering procedure, as in [12]. However, since we usually have several outlier matches (and hence, outlier hypotheses) due to our soft matching constraints, this is not very beneficial and might easily result in outlier hypotheses being clustered together. To avoid this, we want to find the most plausible 3D hypothesis $h_{i,n}^*$ for each 2D line segment $l_{i,n}$, based on the spatial proximity among all its 3D hypotheses. To find this hypothesis, we first define a hypotheses subset $H_{i,n} = \{h_{i,j}^{n,m} \in H\}$ for each 2D segment $l_{i,n}$. Furthermore, we compute a subset $\phi_{i,n}^{r_i}$ for each hypothesis $h \in H_{i,n}$ by

$$\phi_{i,n}^{r_i}(h) = \left\{ \hat{h} \in H_{i,n} \quad | \quad d_{3D}(h,\hat{h}) < r_i \right\}, \tag{3}$$

where $d_{3D}(h,\hat{h})$ stands for the maximum distance in space between two hypotheses h and $\hat{h}$, and r_i denotes a local spatial regularization threshold with respect to image I_i. The subset

$\phi_{i,n}^{r_i}(h)$ can be seen as a nearest neighbor set for h, which includes all other hypotheses from $H_{i,n}$ that are within a certain spatial radius r_i, and h itself (since $d_{3D}(h,h) = 0$). Motivated by [12], we estimate r_i by shifting the 2D segment $l_{i,n}$ (corresponding to a hypothesis h) by a fixed value σ in the image space, and calculate the maximum distance between h and the plane defined by the the camera center of C_i and the camera rays through the shifted segment endpoints. We do the same for the second segment, $l_{j,m}$, and define $r(h)$ to be the average of these measurements. We then take the median over all $r(h)$ for which h references I_i, to obtain the local spatial regularization threshold r_i. We now compute the most plausible hypothesis $h_{i,n}^*$ for $l_{i,n}$ as

$$h_{i,n}^* = \underset{h \in H_{i,n}}{\operatorname{argmax}} \left(p(\phi_{i,n}^{r_i}(h)) \right), \tag{4}$$

where $p(\phi_{i,n}^{r_i}(h))$ is a simple counting function, which returns the number of different cameras supporting the hypothesis subset $\phi_{i,n}^{r_i}(h)$. We denote this as *potential cluster size* (PCS). This procedure is based on the idea that triangulated segments corresponding to correct matches will always be close in space, while wrongly estimated 3D segments can be at an arbitrary position. Hence, at the correct position we can connect the largest number of cameras with minimum effort.

We can now define the set of clusterable hypotheses $H^* \subset H$, which includes $h_{i,n}^*$ for each line segment $l_{i,n}$. To verify clustered hypotheses later on, we require a validity criterion based on a predefined *minimal cluster size* α. This parameter specifies how many different cameras must support a hypotheses cluster to be considered as valid. Since clusters with less than α cameras can never be valid, we only consider a hypothesis $h_{i,n}^*$ for clustering if its PCS is at least α. We call this procedure *local hypothesis selection*, because the selection is only based on the hypotheses set $H_{i,n}$, which only holds hypotheses resulting from valid pairwise matches with respect to the 2D segment $l_{i,n}$. Hence, it does not consider the global hypotheses constellation. This speeds-up the selection process and also prevents that wrong hypotheses for other segment pairs influence the decision procedure for $l_{i,n}$.

2.3 Hypotheses Clustering

Now that we have a set of 3D line segment hypotheses, which are obtained from pairwise 2D line segment matches, we want to cluster corresponding segments together and simultaneously remove remaining outlier hypotheses. To prevent a greedy direct clustering procedure, which does not reflect the global constellation of the 3D hypotheses, we propose to use a graph based clustering approach.

Graph-based clustering requires a pairwise affinity matrix, consisting of similarities between the 3D line hypotheses. In our case, we exploit spatial proximity as similarity measure. Since we may not have a proper scale information in our reconstruction pipeline, we use the spatial regularization thresholds r_i (obtained from the image space) and the so called *span* ($s_\alpha(h) < r_i$) of a hypothesis to define pairwise affinities between hypotheses. The span is basically a local distance measure, which quantifies the uncertainty of a 3D hypothesis based on its neighboring hypotheses. It is defined as the minimum spatial distance we would have to go to directly cluster together enough hypotheses, such that the number of contributing cameras is at least α. Note that this is not a parameter to be chosen but which defines itself based on the uncertainty of the triangulation and the matching scores. We now compute a set D which holds all potentially matching hypotheses tuples as

$$D = \left\{ (h_{i,n}^*, h_{j,m}^*) \quad | \quad M_{i,j}(n,m) = 1 \wedge d_{3D}(h_{i,n}^*, h_{j,m}^*) < r_i \right\}. \tag{5}$$

This ensures that we only compute the spatial distances between hypotheses which contain residuals that are matching in image space as well. We can now transform this set of pairwise matches into an affinity matrix A

$$A(\gamma(i,n),\gamma(j,m)) = \begin{cases} 1 & \text{if } d < s_\alpha(h^*_{i,n}) \\ \left(\frac{r_{\gamma(i,n)}-d}{r_{\gamma(i,n)}-s_\alpha(h^*_{i,n})}\right) & \text{if } s_\alpha(h^*_{i,n}) \leq d < r_{\gamma(i,n)} \\ 0 & \text{else} \end{cases} \tag{6}$$

where each row and column correspond to a hypothesis h^* which is part of at least one tuple in D. The function $\gamma(\cdot)$ is a mapping function which assigns row and column indices for A uniquely to hypotheses in D, and $d = d_{3D}(h^*_{i,n}, h^*_{j,m})$. Since it might be possible that different 2D line segments $l_{i,n}$ (from possibly different views I_i) have the same 3D hypothesis as their personal best, we define the spatial regularization threshold for this hypothesis $r_{\gamma(i,n)}$ as the average over all values r_i. With this information at hand, we apply the efficient graph based clustering procedure by Felzenszwalb and Huttenlocher [8]. Despite the fact that their approach was originally designed for segmentation purposes, it can be used for any kind of clustering where pairwise affinities can be computed [7]. The method is completely unsupervised and automatically identifies the number of clusters. The only required user input is a region preference parameter. Since we do not want to put any a priori restriction on the cluster sizes, we set this value to 2.

After clustering, we merge all hypotheses within a detected valid cluster (number of supporting cameras is at least α). This is done by using the PCA based approach introduced in [11, 12, 16]. Since we do not want to merge collinear segments in 3D to one large segment (because the gaps between them might be physically reasoned), we evaluate the individual 3D segments along the line based on the 2D observations in the image space.

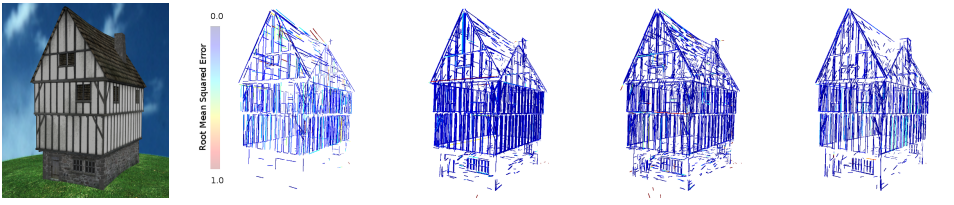
Up to this point we have assumed that all images and camera poses are given right from the start. As a final step, we discuss how our proposed method can be used to create incremental results.

2.4 Incremental Reconstruction

Since we want to integrate our method into any given incremental SfM system, we have to start from an empty image set I. We then simply perform all aforementioned steps whenever a new image I_i and camera pose C_i is provided by the underlying SfM pipeline. At the beginning (when $|I| < \alpha$) it does not make sense to perform the hypothesis selection and clustering procedure, but the matching and triangulation can off course be done. Given that we only use the M nearest neighbors for the 2D line segment matching, we always have a limited number of hypotheses for clustering (i.e. all selected hypotheses which have residuals in the current scope $I_i \cup \hat{I}_M(i)$).

Since we already have a partial reconstruction (after several clustering steps when $|I| \geq \alpha$), some of the hypotheses may have already been clustered before. Hence, such hypotheses have more than two residuals and also more than two triangulated segments. This information is only used during the hypotheses merging (after the clustering), but not during the hypothesis selection. This is due to the fact that the potential cluster size is computed for each contributing 2D segment individually, which does not take the actual cluster size of a hypothesis into account. We chose this representation to allow the system to correct wrong clusters at a later point, e.g. when more information is available.

To keep the per image processing time approximately constant (especially for large image sequences), we have to remove unpromising hypotheses from time to time. Therefore, we



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

Figure 3: Reconstruction results for the *Timberframe* sequence (240 images). (a) Example image. (b) The original result by [16] (offline, runtime of several hours), RMSE = 0.291. (c) The result by [11] (offline, runtime of 45 minutes), RMSE = 0.094. (d) The result by [12] (online, 5.7 minutes) RMSE = 0.196. (e) Our reconstruction with appearance and collinearity constraints enabled (online, 6.9 minutes), RMSE = 0.095.

compute a condition number $\rho(i)$ for each image I_i, which holds the number of matched visual neighbors that have been added to the reconstruction after I_i. We remove invalid hypotheses for which the average condition number among their supporting views is bigger than α.

3 Experimental Results

We evaluate our proposed method on several challenging datasets. As line segment detector we use the LSD algorithm [26], and as incremental SfM pipeline we use [13] (which ensures a fair comparison to the the only existing incremental method [12]). The test system is a standard desktop PC. All parameters are initialized with default values and remain unchanged for all experiments. For the appearance-based matching refinement we set the smoothness parameter $k = 7$, $t_{sim} = 0.5$, and the number of bins is 166 (see [3]). Furthermore, $M = 10$ and $\alpha = 4$. The only parameters which need adaptions are the uncertainty σ, and the collinearity threshold σ_c. To use one value for all experiments we scale the images down to a fixed size (FullHD resolution). Choosing a higher value for σ is especially beneficial for the estimation of the best hypothesis, since the probability of missing some information is decreased. Though, choosing σ too big may lead to wrong estimates, since outliers may gain more importance. We therefore set $\sigma = 10$ by default. Since it does not make any sense to set $\sigma_c > \sigma$, we also fix $\sigma_c = \sigma$.

3.1 Experiments

To demonstrate the capabilities of our proposed algorithm, we performed several quantitative and qualitative experimental evaluations. As a quantitative evaluation we used the synthetic *Timberframe* [1] dataset from [16], since there is a groundtruth CAD model available. Figure 3 shows our result in comparison to related state-of-the-art methods [11, 12, 16].

As can be seen, our proposed method achieves more accurate results than a previous incremental approach [12] (RMSE 0.095 vs. 0.196), while the runtime is not largely increased (6.9 vs. 5.7 min). That is off course due to the non-greedy nature of our approach and the incorporation of collinearity information. The accuracy with respect to the ground truth CAD model is almost as high as for the offline approach [11] (RMSE 0.095 vs. 0.094), which achieves the highest accuracy among the competitive algorithms, but with a significantly higher processing time (6.9 vs 45 min).

[1] http://www.mpi-inf.mpg.de/resources/LineReconstruction

	[12], $\sigma = 1$				[12], $\sigma = 5$				Ours			
Sequence	#clus.	#seg.	∅res.	∅t	#clus.	#seg.	∅res.	∅t	#clus.	#seg.	∅res.	∅t
PYLON	1281	1281	5.5	1.8	2657	2657	8.6	1.6	1075	1346	**21.9**	1.5
HOUSE	1524	1524	6.4	1.2	2268	2268	9.3	1.1	483	991	**41.9**	1.8
EIFFEL	204	204	5.0	0.9	943	943	8.4	0.8	283	366	**20.2**	1.1
TIMBER	2355	2355	7.8	1.4	3436	3436	14.9	1.2	1342	1866	**30.9**	1.4

Table 1: Relevant numbers for the used test sequences. *#clus.* stands for the number of 3D clusters, *#seg.* is the number of individual 3D line segments, *∅res.* is the average number of residuals, and *∅t* is the average computing time per image (in seconds).

We further qualitatively compare our approach to [12] on three challenging test sequences. The reconstruction results are shown in Figure 4, the relevant numbers (e.g. runtime, residuals, ...) in Table 1 (including the *Timberframe* dataset from above). We compare our method to two versions of [12] with $\sigma = 1$ (default) and $\sigma = 5$, which roughly corresponds to the spatial distance for which the affinities in our approach are above 0.5. As can be seen, our method produces much cleaner results while the runtime is not significantly increased (around 0.2 sec per image on average). The most significant improvement is the very high number of average residuals per cluster (highlighted in Table 1). As can be seen, our method manages to create much bigger clusters, which explains why the number of individual 3D segments is usually much lower compared to [12] with a similar clustering distance (even though no relevant parts of the objects are missing). This is off course also a result of the incorporation of collinearity information.

A demonstration how the various steps of the matching procedure affect the results and the runtime is shown in Figure 5. As can be seen, the visual appearance of the results does not vary significantly while both runtime and average residual number are strongly affected. The usage of collinearity information enables us to create much bigger clusters (approx. 50% bigger), while the runtime can be significantly higher (especially when no appearance information is used). The usage of both collinearity and appearance information is in general a good compromise between runtime and completeness of the obtained 3D models.

4 Conclusion

We have proposed a novel method to generate incremental 3D line models based on the output of an online SfM pipeline. We have shown that using a semi-global approach rather than direct greedy clustering significantly improves the accuracy of the obtained models, with approximately the same runtime. Furthermore, the usage of appearance and collinearity information improves the matching results even when wiry structures are to be reconstructed. The relaxation of the uncertainty parameter σ allows to generate more complete 3D models with the same (or even less) images, which is especially beneficial when the proposed method is used within a time-critical SLAM [6] system (e.g. to create trackable 3D line models on-the-fly). In the near future, we want to investigate the usage of other edge-based features (e.g. curves [19]) since not all object boundaries can be approximated with line segments.

Acknowledgements

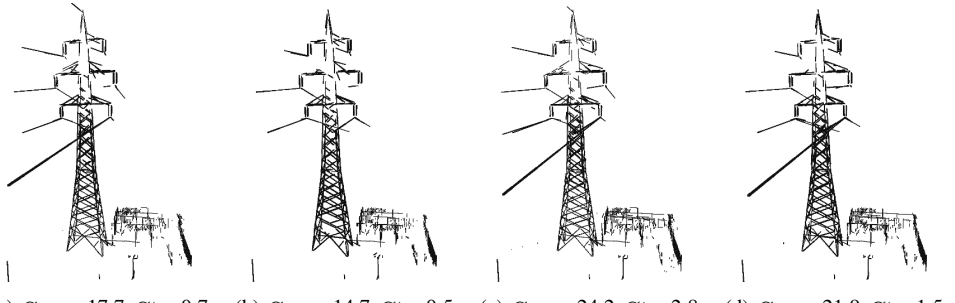
This work has been supported by the Austrian Research Promotion Agency (FFG) project FreeLine (Bridge1/843450) and OMICRON electronics GmbH.

(a) [▭], $\sigma = 1$

(b) [▭], $\sigma = 5$

(c) Ours

Figure 4: Reconstruction results for the PYLON [▭, ▭], HOUSE and EIFFEL sequences.



(a) $\varnothing$res = 17.7, $\varnothing$t = 0.7 (b) $\varnothing$res = 14.7, $\varnothing$t = 0.5 (c) $\varnothing$res = 24.2, $\varnothing$t = 2.8 (d) $\varnothing$res = 21.9, $\varnothing$t = 1.5

Figure 5: Reconstruction results for the PYLON sequence. (a) Geometric matching only, (b) Geometric + Appearance, (c) Geometric + Collinearity, (d) Geometric + Appearance + Collinearity.

References

[1] S. Agarwal, N. Snavely, I. Simon, and S.M. Seitz. Building rome in a day, 2009. International Conference on Computer Vision (ICCV).

[2] A. Bartoli, M. Coquerelle, and P. Sturm. A framework for pencil-of-points structure-from-motion, 2004. European Conference on Computer Vision (ECCV).

[3] H. Bay, V. Ferrari, and L. van Gool. Wide-baseline stereo matching with line segments, 2005. International Conference on Computer Vision and Pattern Recognition (CVPR).

[4] F. Bin, Wu. Fuchao, and Hu. Zhanyi. Line matching leveraged by point correspondences, 2010. International Conference on Computer Vision and Pattern Recognition (CVPR).

[5] F. Bin, Wu. Fuchao, and Hu. Zhanyi. Robust line matching through line-point invariants, 2011. Pattern Recognition.

[6] A.J. Davison. Real-time simultaneous localization and mapping, 2002. International Conference on Computer Vision (ICCV).

[7] M. Donoser. Replicator graph clustering, 2013. British Machine Vision Conference (BMVC).

[8] P. Felzenszwalb and F. Huttenlocher. Efficient graph-based image segmentation, 2004. International Journal of Computer Vision (IJCV).

[9] J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, and M. Pollefeys. Building rome on a cloudless day, 2010. European Conference on Computer Vision (ECCV).

[10] K. Hirose and H. Saito. Fast line description for line-based SLAM, 2012. British Machine Vision Conference (BMVC).

[11] M. Hofer, A. Wendel, and H. Bischof. Line-based 3D reconstruction of wiry objects, 2013. Computer Vision Winter Workshop (CVWW).

[12] M. Hofer, A. Wendel, and H. Bischof. Incremental line-based 3D reconstruction using geometric constraints, 2013. British Machine Vision Conference (BMVC).

[13] C. Hoppe, M. Klopschitz, M. Rumpler, A. Wendel, S. Kluckner, H. Bischof, and G. Reitmayr. Online feedback for structure-from-motion image acquisition, 2012. British Machine Vision Conference (BMVC).

[14] C. Hoppe, M. Klopschitz, M. Donoser, and H. Bischof. Incremental surface extraction from sparse structure-from-motion point clouds, 2013. British Machine Vision Conference (BMVC).

[15] K. Hyunwoo and L. Sukhan. A novel line matching method based on intersection context, 2010. International Conference on Robotics and Automation (ICRA).

[16] A. Jain, C. Kurz, T. Thormaehlen, and H. Seidel. Exploiting global connectivity constraints for reconstruction of 3D line segments from images, 2010. International Conference on Computer Vision and Pattern Recognition (CVPR).

[17] B. Khaleghi, M. Baklouti, and F.O. Karray. SILT: Scale-invariant line transform, 2009. Computational Intelligence in Robotics and Automation (CIRA).

[18] D. Lowe. Distinctive image features from scale-invariant keypoints, 2004. International Journal of Computer Vision (IJCV).

[19] V. Patraucean, P. Gurdjos, and R.G. von Gioi. A parameterless line segment and elliptical arc detector with enhanced ellipse fitting, 2012. European Conference on Computer Vision (ECCV).

[20] D. Pollefeys, M. Nister, J.-M. Frahm, A Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S.J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewenius, R. Yang, G. Welch, and H. Towles. Detailed real-time urban 3D reconstruction from video, 2008. International Journal of Computer Vision (IJCV).

[21] M. Pollefeys, L. van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera, 2004. International Journal of Computer Vision (IJCV).

[22] G. Schindler, P. Krishnamurthy, and F. Dellaert. Line-based structure from motion for urban environments, 2006. International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT).

[23] N. Snavely, S.M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D, 2006. ACM SIGGRAPH.

[24] N. Snavely, S.M. Seitz, and R. Szeliski. Modeling the world from internet photo collections, 2008. International Journal of Computer Vision (IJCV).

[25] B. Verhagen, R. Timofte, and L. van Gool. Scale-invariant line descriptors for wide baseline matching, 2014. Winter Conference on Applications of Computer Vision (WACV).

[26] R.G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. LSD: A fast line segment detector with a false detection control, 2010. Transactions on Pattern Analysis and Machine Intelligence (PAMI).

[27] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof. Dense reconstruction on-the-fly, 2012. International Conference on Computer Vision and Pattern Recognition (CVPR).

[28] C. Wu. Towards linear-time incremental structure from motion, 2013. International Conference on 3D Vision (3DV).

[29] L. Zhang and R. Koch. Line matching using appearance similarities and geometric constraints, 2012. Lecture Notes in Computer Science: Pattern Recognition.

[30] Y. Zhang, H. Yang, and X. Liu. A line matching method based on local and global appearance, 2011. International Congress on Image and Signal Processing (ICISP).

[31] W. Zhiheng, W. Fuchao, and H. Zhanyi. MSLD: A robust descriptor for line matching, 2009. Pattern Recognition.