SINGLE-SHOT DENSE RECONSTRUCTION WITH EPIC-FLOW

Qiao Chen[*]*, Charalambos Poullis*[†]

Immersive and Creative Technologies Lab,
Concordia University

ABSTRACT

In this paper we present a novel method for generating dense reconstructions by applying only structure-from-motion(SfM) on large-scale datasets without the need for multi-view stereo as a post-processing step. A state-of-the-art optical flow technique is used to generate dense matches. The matches are encoded such that verification for correctness becomes possible, and are stored in a database on-disk. The use of this out-of-core approach transfers the requirement for large memory space to disk, therefore allowing for the processing of even larger-scale datasets than before. We compare our approach with the state-of-the-art and present the results which verify our claims.

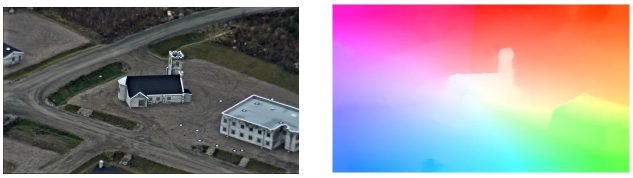
Index Terms — 3D reconstruction, dense reconstruction, structure -from-motion (SfM), multi-view stereo (MVS), urban reconstruction, large-scale

1. INTRODUCTION

The automatic reconstruction of large-scale urban areas has always been of great interest to the computer graphics and vision communities. Image-based reconstructions rely on structure from motion (SfM) to recover the camera poses using bundle adjustment [1, 2, 3], followed by multi-view stereo (MVS) [4, 5] to generate a dense pointcloud. In recent years, many variants of these techniques have been proposed which result in impressive reconstructions.

However, dealing with remote sensor images covering large-scale areas introduces certain challenges which very often cause failures in SfM and/or MVS techniques. Firstly, remote sensor images cover large areas which contain thousands of geospatial features e.g. buildings, roads, trees, cars, etc, which from an oblique aerial or nadir direction look identical and repetitive i.e. consider a satellite image where the roads, the roofs of the building, etc, have the same texture and similar shapes. One of the main limitations of existing state of the art feature extraction and matching techniques is that they cannot handle repetitive textures, leading to erroneous matches and subsequently erroneous or failed reconstructions. Secondly, remote sensor images typically have a large size and capture the object from all around similar to an inverted turn-table i.e. consider the single frame in Figure 1a part of a video captured from a helicopter circling the church building. The symmetry occurring in man-made structures such as this one often leads to erroneous results since features from opposing sides of the building can be easily mistakenly matched i.e. the cameras are facing each other.

In this paper, we propose a method for single-shot dense reconstruction using only SfM. Unlike existing techniques, we rely on the state of the art optical flow technique EpicFlow [6] to extract robust dense matches. The matches are encoded and stored on-disk therefore transferring the requirement for large memory to disk which is easily met. An advantage of the encoding is the fact that verification for correctness can be easily performed and ambiguous matches for which the transitivity property fails are removed. This process is explained in Section 3.1. An iterative bundle adjustment is used which allows for the optimization of an arbitrary number of parameters as explained in Section 3.2. Finally, Section 4 presents the experiments and comparisons with other state of the art techniques which verify our claims.



(a) (b)

Figure 1: (a) A frame from a video captured from a helicopter circling a church building. (b) Epic-flow of two consecutive frames.

Our technical contributions are:

- A novel method of generating dense reconstructions using only SfM while producing similar or better results with other state of the art, in terms of accuracy and time.
- An encoding for the matches which allows the easy identification and elimination of ambiguous matches for which the transitivity property does not hold. This ensures the robustness of the dense matches used for SfM.

2. RELATED WORK

We present related work in terms of (a) dense matching and, (b) 3D reconstruction.

2.1. Dense Matching

Optical flow is the apparent motion between two consecutive frames caused by the movement of the object or the camera. A number of different robust techniques have already been proposed for recovering the optical flow which can be better categorized in terms of the underlying technique they use i.e. block-matching, feature tracking, and energy-based methods. Differential methods of estimating optical flow are based on computing the partial derivatives of the image and the flow field, such as LucasKanade or BuxtonBuxton [7]. The majority of current optical flow methods strongly resemble the original formulation of Horn-Schunck [8]. They combine a data term that assumes constancy of some image property with a spatial term that models how the flow is expected to vary across the image. Current state-of-the-art can be better categorized as follows: coarse-to-fine estimation to deal with large motions [9], texture decomposition [10] or high-order

[*]cq.jocelyn@gmail.com
[†]charalambos@poullis.org

filter constancy [11] to reduce the influence of lighting changes, warping with bicubic interpolation [12], graduated non-convexity to minimize non-convex energies [13], median filtering after each incremental estimation step to remove outliers [14]. FlowNet2.0 [15] re-casts the optical flow estimation as a learning problem and make an improvement over learning optical flow in terms quality and speed.

Perhaps the most popular state of the art optical flow technique which has already been used in many successful vision systems is Epic-Flow[6]. Epic-flow is an edge-preserving interpolation of correspondences for optical flow which leverages recent advances in matching algorithms and introduces an edge-aware geodesic distance that handles motion discontinuities and occlusions. We choose to use this method to compute our dense matches because of its ability to handle deformations and repetitive textures.

2.2. 3D Reconstruction

Given a set of matches, SfM can produce a sparse recostruction of the scene. Typically SIFT is used for detecting and matching features [16] followed by camera pose estimation [17, 1], and finally bundle adjustment [1, 18, 3]. Modern SfM approaches showed great success in reconstructing 3D models for large scale areas from community photo collections shared on the internet, such as in [1, 5]. Another variant is incremental SfM which surpasses traditional SfM techniques in terms of robustness, accuracy, completeness, and scalability.

Perhaps the closest work to ours is COLMAP [3] where a general-purpose SfM system is proposed which incorporates an iterative bundle adjustment, retriangulation, and an outlier filtering strategy that improves completeness and accuracy for large scale datasets.

3. METHODOLOGY

Image matches extracted using dense optical flow are encoded, verified for correctness, and stored in a database as explained in Sections 3.1, 3.1.1, 3.1.2, respectively. The reconstruction is performed using the matches as explained in Section 3.2.

3.1. Pre-processing

During the pre-processing step dense features are extracted and matched between the images. An out-of-core process performs redundancy checks in order to eliminate ambiguous matches [group of matches where the transitive relation does not hold, duplicates] and transforms the validated data into the internal representation used. Finally, the data is encoded and used to populate the database.

3.1.1. Image Matching

Dense features are extracted and matched in an N^2 fashion (complexity is $(N-1) \times (N-2)$) between pairs of images. Although any dense feature extractor/matching technique can be used e.g. SiftFlow [19], FlowNet2,0[15], etc, we employed Epic-flow [6]. Epic-flow computes dense optical flow using a hierarchical, multilayer, correlational architecture inspired by deep convolutional networks even in the presence of large displacements. This matching algorithm can handle complex cases such as non-rigid deformations and repetitive textures, it efficiently determines dense correspondences in the presence of significant changes between images, and it has bidirectional validity checks.

3.1.2. Feature Match Encoding and Verification

Bundle adjustment is the common method for solving Structure-from-Motion problems. Perhaps the most popular variant of this method is the Sparse Bundle Adjustment which exploits the sparse nature of the matrix to efficiently store, process, and solve for relatively large sets of parameters. This variant requires that all information about the matches, the 3D points corresponding to those matches, and the camera parameters are available in memory. When used with a sparse set of matches such as those produced by SIFT [16], SURF [20], ORB [21], etc, this does not constitute a problem however, there is always an upper bound on the number of parameters one can solve for and is typically restricted to a finite set of sparse features.

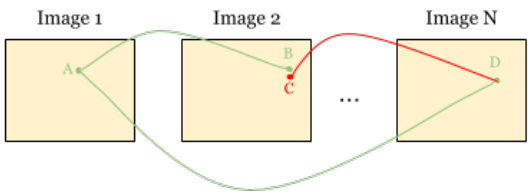


Figure 2: Transitivity property

The second variant of bundle adjustment uses an iterative approach where a non-linear optimization is used to solve for the unknown camera and structure parameters. Until recently this method also required that all information is available in memory which again limited the size of datasets one could process. In COLMAP [3], the authors presented for the first time how incorporating a database allows the processing and solving of larger sets of parameters however, this was again limited to a set of sparse features, though albeit larger than before, but which almost always contained ambiguous matches.

In order to address these problems and ensure that only validated and unambiguous dense matches are used we represent the data in a format which allows for the efficient identification of redundancies. By redundancies we refer to (a) duplicate matches, and (b) matches where the transitivity property does not hold as shown in Figure 2, if 2 or more feature points in one image are matched to the same feature point or its match, they will be removed. We achieve this by keeping two maps for each image in the dataset, an index map, and a conflict map. The conflict maps keep track of whether a feature point has a match in the next image, and the index maps store the information about the indices of feature points in each of the feature tracks. This reduces the complexity of checking a feature point whether it is ambiguous, to 1. We render images for verification from verified matches of its previous frame, as shown in the example in Figure 3b, and compare to its original image, as shown in the example in Figure 3a.

(a) (b)

Figure 3: (a) A frame from our dataset (b) A rendered image from verified matches of its previous frame

3.1.3. Populating the Database

The internal data representation and redundancy check ensures that there are no duplicates and that for all matches the transi-

tivity property holds. Next, we populate the database using this information. To speed up the recall time we encode the information as a single number and use a single table for storage instead of multiple tables [3]. This eliminates complex queries involving joins which are computationally expensive.

A feature $f_{(i,x,y)}$ contained in image I_i at pixel (x, y) is encoded as a single number $e_{(i,x,y)} = i * w * h + y * w + x$, where w, h are the width and height of the image respectively. Similarly the decoding of a number into the three tuple is performed as $x = code\%w, y = ((code - x)\%(w * h))/(w), i = (code - x - y * w)/(w * h)$, where (x, y) are the image coordinates and i is the image index.

3.2. Bundle Adjustment

Bundle adjustment involves the simultaneous optimization of 3D points and camera poses based on the reprojection error. Using an initial estimate for the camera poses from the decomposition of the fundamental matrix between pairs of images, the initial 3D points are estimated via triangulation. The optimization proceeds by updating the 3D points and camera poses such that the reprojection error E is minimized [1] given by,

$$E = \sum_i d_i(\|Q(C_c, X_k), x_i\|)^2 \quad (1)$$

where C_c are the camera parameters, X_k the points, and $Q(.,.)$ is a function which projects a 3D point onto the image plane corresponding to camera parameters C_c. d_i is a loss function which potentially down-weights outliers. A popular method for solving this type of problems is to store and factor the data as a dense sparse matrix or apply a non-linear optimization using Levenberg-Marquardt. Solving using a dense or sparse matrix requires N^2 memory space and has complexity of $O(N^3)$ however for large-scale datasets it very often fails due to the large memory requirements imposed. On the other hand, solving using the iterative method has $O(N)$ time complexity and requires N memory space. However the memory requirement can be reduced by computing Equation 1 in batches.

Inspired by COLMAP [3] and retriangulation, we use the iterative bundle adjustment method, since the generated dense flow between pairwise images leads to vast amount of points. This scheme is more efficient in our case, because the number of cameras is much smaller than the number of points, and we avoid performing large-scale matrix computations in memory.

4. EXPERIMENTS

We run experiments on a dataset containing images taken from a helicopter circling around a church building. Our test dataset contains 71 images with resolution 1280×720 with unknown camera calibrations or EXIF information. We use the same dataset to evaluate our proposed method and compare it to the state of the art incremental SfM techniques, namely Bundler [1], VisualSFM [2], COLMAP [3]. The reconstructions are compared and evaluated by computing the distance of points set, as well as performing surface reconstruction and comparing the meshes.

One of the most popular feature extraction methods in Structure from Motion is SIFT [16], in COLMAP [3], all experiments use RootSIFT features and match each image. With our dataset, each pair of the matched images have less than 1000 feature matches using SIFT, there are twice as many using RootSIFT. However, in our case, each of the images has almost the same amount of feature points as the resolution in the second image, thus an on-disk database for computing feature tracks becomes essential because

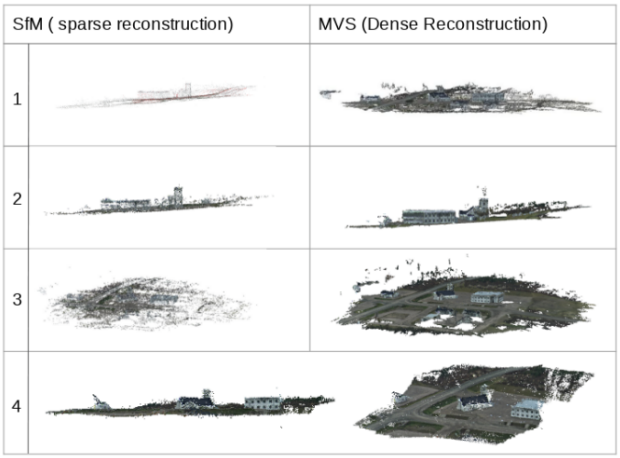


Figure 4: A comparison of results: 1. sparse reconstruction of SfM (left) and dense reconstruction of PMVS (right) ; 2. Side-view of COLMAP SfM (left) and surface reconstruction (right) 3. Top-view of COLMAP SfM (left) and surface reconstruction (right) 4. Side-view and top-view of direct SfM result ours

of memory restrictions. Each of the rows in our database is converted to one feature track, then with COLMAP's [3] iterative bundle adjustment we generate dense 3D reconstruction using only SfM and in a shorter time, as shown in Table 1. Rather than having a two steps process, i.e. SfM + MVS, which is time consuming and restricted by memory limitations, the proposed method generates a dense model efficiently and directly from SfM.

Figure 4 shows a comparison our result with the results of both SfM sparse and, MVS dense reconstructions of state of the art VisualSFM[2] and, COLMAP[3].

We also performed a comparison between the sparse point cloud produced by our approach and COLMAP[3] by computing the nearest distance between points, and computing the mean distance of 0.01089, RMS 0.04825, with an overlap of 80%. Another comparison was performed between the reconstructed surfaces of the dense point clouds and computing the Hausdorff Distance of the two meshes.

Bundler [2] generated disjoint groups of images (based on the matches) which led to multiple reconstructions of different scales, therefore, we were unable to quantitatively compare the results because considerable user interation is required to manually aligned the reconstructions which introduced errors/bias.

Finally, We compared our reconstructed surface to Kazhdan's [22] and computed a mean distance of 0.013398 and RMS of 0.021225, respectively.

As it can be seen, the proposed approach produces similar or better reconstructions (in terms of accuracy and density) with the dense techniques at a fraction of the time, using only a single step of SfM. It produces better results than all techniques (sparse or dense) except from [22] which takes three times longer to generate a result.

5. CONCLUSION

In this paper we have presented an improved method for performing single-shot reconstructions for large-scale datasets using only SfM. The method relies on a state of the art optical flow technique to generate robust matches. The matches are further refined by verifying the correctness. An iterative bundle adjustment method is used to reconstruct the scene which is similar or better than other dense reconstruction state of the art techniques.

Method	Features	# matches	SfM-Variant	# points	time-SfM(min)	MVS-Variant	# points	time-MVS(min)	Run-time
Visu-alSFM	SIFT	-	Bundler [2]	4,863	1.10	PMVS	111,189	1.289	2.389
COLMAP	RootSIFT	704,127	iterative BA [3]	11,274	11.051	Kazhdan [22]	287,205	23.965	35.016
Our method	Epic-Flow [6]	1,576,705	iterative BA [3]	**139,606**	**12.533**	-	-	-	**12.533**

Table 1: The comparison of number of points reconstructed and runtime. Although our method does not produce as many points as state of the art COLMAP-Kazhdan [22] (dense), it produces more points than COLMAP-SfM-sparse by a factor of 10, more points than PMVS

Acknowledgment

This research is based upon work supported by the Natural Sciences and Engineering Research Council of Canada Grants No. N01670 (Discovery Grant) and DNDPJ515556-17 (Collaborative Research and Development with the Department of National Defence Grant).

6. REFERENCES

[1] D Gallup JM Frahm, P Fite-Georgel, "Building rome on a cloudless day(2010)," *Daniilidis K., Maragos P., Paragios N. (eds) Computer Vision ECCV 2010. ECCV 2010. Lecture Notes in Computer Science*, vol. vol 6314. Springer, Berlin, Heidelberg, 2010.

[2] Changchang Wu, "Towards linear-time incremental structure from motion," *3D Vision - 3DV 2013, 2013 International Conference*, 2013.

[3] Johannes L. Schonberger and Jan-Michael Frahm, "Structure-from-motion revisited," *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Schonberger 2016 CVPR*, 2016.

[4] Tomas Pajdla Daniel Martinec, "Robust rotation and translation estimation in multiview reconstruction," *Computer Vision and Pattern Recognition, CVPR '07. IEEE Conference*, 2007.

[5] Jean Ponce Yasutaka Furukawa, "Accurate, dense, and robust multi-view stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 32, Issue: 8, Aug. 2010*.

[6] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid, "EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow," 2015.

[7] B. F. Buxton and H. Buxton, "Computation of optic flow from the motion of edge features in image sequences," *Image and Vision Computing*, vol. 2(2):59-75, 1942.

[8] Brian G.Schunck Berthold K.P.Horn, "Determining optical flow," *Int. J. Comput. Vision 61 (3) (2005) 211231*.

[9] N. Papenberg T. Brox, A. Bruhn and J. Weickert., "High accuracy optical flow estimation based on a theory for warping," *European Conference on Computer Vision, pages 2536, 2004.*

[10] J. Braun U. Franke A. Wedel, T. Pock and D. Cremers., "Duality tv-l1 flow with fundamental matrix prior," *Image and Vision Computing New Zealand, 2008*.

[11] S. Roth V. Lempitsky, C. Rother and A. Blake., "Fusion moves for markov random field optimization," *IEEE Transaction on Pattern Analysis Machine Intelligence, 32(8):13921405, August 2010*.

[12] S. Roth V. Lempitsky and C. Rother., "Fusionflow: Discrete-continuous optimization for optical flow estimation.," *Image and Vision Computing, 2010.*

[13] J. P. Lewis D. Sun, S. Roth and M. J. Black., "Learning optical flow," *In European Conference on Computer Vision, pages 8397, 2008*.

[14] C. Zach D. Cremers A. Wedel, T. Pock and H. Bischof., "An improved algorithm for tv-l1 optical flow," *In Dagstuhl Motion Workshop, pages 2345, 2008*.

[15] Tonmoy Saikia Margret Keuper Alexey Dosovitskiy Thomas Brox Eddy Ilg, Nikolaus Mayer, "Flownet 2.0: Evolution of optical flow estimation with deep networks," *Computer Vision and Pattern Recognition, 2017 IEEE Computer Society Conference*.

[16] David G. Lowe, "Object recognition from local scale-invariant features," *ICCV '99 Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*.

[17] D. Martinec and T. Pajdla., "Robust rotation and translation estimation in multiview reconstruction," *Computer Vision and Pattern Recognition, 2007. CVPR07. IEEE Conference on, pages 18. IEEE, 2007.*

[18] B. Curless C. Wu, S. Agarwal and S. M. Seitz., "Multicore bundle adjustment," *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[19] Jenny Yuen C. Liu, "Sift flow: Dense correspondence across scenes and its applications," *Journal IEEE Transactions on Pattern Analysis and Machine Intelligence archive Volume 33 Issue 5, May 2011 Pages 978-994*.

[20] Van Gool L. Bay H., Tuytelaars T., "Surf: Speeded up robust features," *Leonardis A., Bischof H., Pinz A. (eds) Computer Vision ECCV 2006. ECCV 2006. Lecture Notes in Computer Science, vol 3951. Springer, Berlin, Heidelberg*, 2006.

[21] K. Konolige E. Rublee, V. Rabaud and G. Bradski., "Orb: An efficient alternative to sift or surf," *Computer Vision (ICCV), 2011 IEEE International Conference on, 2011.*

[22] Jan-Michael FrahmMarc Pollefeys Johannes L. Schönberger, Enliang Zheng, "Pixelwise view selection for unstructured multi-view stereo," *European Conference on Computer Vision ECCV 2016: Computer Vision ECCV 2016 pp 501-518.*