BLOCK-BASED MOTION ESTIMATION SPEEDUP FOR DYNAMIC VOXELIZED POINT CLOUDS

Camilo Dorea and Ricardo L. de Queiroz

Department of Computer Science
University of Brasilia, DF, Brazil
Email: camilodorea@unb.br, queiroz@ieee.org

ABSTRACT

Motion estimation is a key component in dynamic point cloud analysis and compression. We present a method for reducing motion estimation computation when processing block-based partitions of temporally adjacent point clouds. We propose the use of an occupancy map containing information regarding size or other higher-order local statistics of the partitions. By consulting the map, the estimator may significantly reduce its search space, avoiding expensive block-matching evaluations. To form the maps we use 3D moment descriptors efficiently computed with one-pass update formulas and stored as scalar-values for multiple, subsequent references. Results show that a speedup of 2 produces a maximum distortion dropoff of less than 2% for the adopted PSNR-based metrics, relative to distortion of predictions attained from full search. Speedups of 5 and 10 are achievable with small average distortion dropoffs, less than 3% and 5%, respectively, for the tested data set.

***Index Terms*—** Point clouds, volumetric media, 3D, motion estimation.

1. INTRODUCTION

Point clouds are one of the new and upcoming means for representing volumetric media in immersive communications. They may be considered as a collection of points (x, y, z) in 3D space with attributes such as color, normals, transparency, specularity, etc. The point clouds are said to be voxelized when points are constrained to lie in a regular 3D grid and assume integer coordinate values. The points within such grid are called voxels and may be occupied or not. Dynamic point clouds depict a sequence of such clouds over time and, like video for images, may display movement. Each point cloud within this temporal sequence constitutes a frame.

Recent efforts have been dedicated to the compression of static point clouds, focusing on compression of geometry [1, 2] as well as color attributes [3, 4]. Compression of dynamic point clouds have also been subject of study [5–7]. In these works, motion estimation (ME) is crucial to performance and may account for significant portions of coding execution time. In [5], ME matches successive clouds using features derived from graph transforms. Other works [6, 7] use block-based partitions of the point cloud and search for corresponding blocks in temporally adjacent clouds, as exemplified in Fig. 1. An iterative closest point (ICP) algorithm is applied in [6] and [7] proposes the optimization of a block-matching metric but does not define a search scenario.

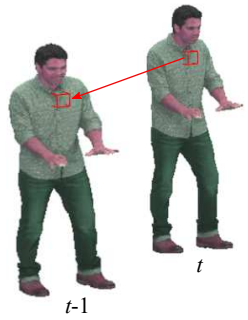
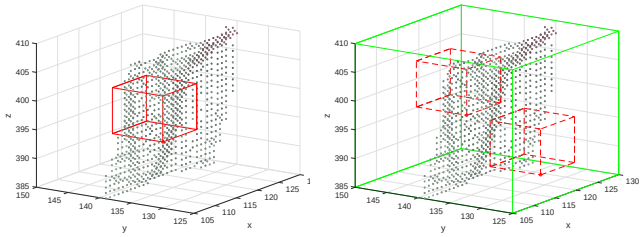


Fig. 1: Examples from the point cloud sequence *Loot* [8] shown at different time instants and perspectives. Motion estimation establishes correspondences between block-based partitions of the clouds.

Block-based ME for point clouds presents similarities with block-matching of conventional 2D video. For each block, assumed as a cube of dimensions $L \times L \times L$, in the current (or source) frame, a search space of size $S \times S \times S$ is defined in the previous (or target) frame around the co-located cube, as depicted in Fig. 2. Among the set of dimension-L target cubes available within the search space, a best match with respect to the source cube is determined. Lastly, all voxels within the selected target cube are compensated with the determined motion vector, i.e., the displacement between matched cubes, forming a prediction of the current frame's point cloud. Differently from conventional 2D ME, however, the search space contains unoccupied voxels and fast matching schemes based on search space sampling, e.g., [9], must operate with prior knowledge of these geometry differences.

This work was partially supported by FAPDF grant 193.001.280/2016 and by CNPq grant 308150/2014-7.



(a) (b)

Fig. 2: A point cloud (a) source cube and (b) candidate target cubes from an adjacent frame shown within a search space.

The baseline for block-based ME is a full search, for each source cube, among all $S \times S \times S$ cubes of its search space. We propose the use of a fast, pre-computed *occupancy map* which succinctly describes spatial distribution within each target cube to speedup this process. By consulting the map and comparing the occupancy value to that of the source cube's, the motion estimator may discard target cubes and avoid evaluation of cube-matching with incompatible candidates. Note that matching criteria often require establishment of voxel-wise geometric and color correspondences and may account for a significant part of computation time. We measure speedup as the number of discarded candidates, with respect to full search, and observe distortion of the predicted frame. As occupancy descriptors we use size (occupied voxel count) as well as other higher-order statistics. These descriptors are quickly calculated by taking advantage of the overlap among target cubes. Furthermore, in our implementation, the map is scalar-valued and may thus be efficiently stored. In our experiments we adopt a simplified version of the cube-matching criterion in [7], briefly described in Sec. 4. Note that alternate matching criteria may be employed in our framework.

We present the 3D moments used for local shape description in Sec. 2 and their usage within occupancy maps for ME in Sec. 3. Experimental results and conclusions are discussed in Secs. 4 and 5.

2. 3D MOMENTS

Image moments, moment invariants and their extension to 3D have been used in analysis and object recognition for many years [10]. In this section we review some basic definitions and cast these into computationally efficient forms applicable to point cloud analysis.

Consider a voxel density function $v(x, y, z)$ which assumes value 1 for occupied and 0 for unoccupied voxels within a domain of size L^3. The 3D moment of order $p+q+r$ may be defined as

$$M_{pqr} = \sum_{x=0}^{L-1} \sum_{y=0}^{L-1} \sum_{z=0}^{L-1} x^p y^q z^r v(x, y, z). \quad (1)$$

Moments provide a measure of the spatial distribution or shape of a set of points. The zeroth order moment M_{000} represents the size of the point cloud. For the sake of notational convenience it will be referred to as M_0. Its centroid $(\bar{x}, \bar{y}, \bar{z})$ is defined as the first order moments divided by size. The variance μ'_{200} (in the x-direction) is specified in terms of the central moment

$$\mu_{pqr} = \sum_{x,y,z} (x - \bar{x})^p (y - \bar{y})^q (z - \bar{z})^r v(x, y, z) \quad (2)$$

such that $\mu'_{200} = \mu_{200}/\mu_{000}$.

When analyzing multiple, overlapping partitions of a point cloud, arbitrary-order statistical moments may be efficiently calculated with one-pass update formulas [11]. Consider, without loss of generality, a cube (as illustrated in Fig. 2) sliding along the x-direction. Each increment in x introduces a set A and discards a set B of L^2 points such that the size, centroid and second order central moment of the next cube C' may be updated in terms of the statistics of the previous cube C as

$$M_0^{C'} = M_0^C + M_0^A - M_0^B, \quad (3)$$

$$\bar{x}^{C'} = (\bar{x}^C M_0^C + \bar{x}^A M_0^A - \bar{x}^B M_0^B)/M_0^{C'} \quad (4)$$

and, adapted from [12],

$$\mu_{200}^{C'} = \mu_{200}^C + \mu_{200}^A - \mu_{200}^B + \frac{(\bar{x}^C - \bar{x}^A)^2 M_0^C M_0^A}{M_0^C + M_0^A} + \frac{(\bar{x}^C - \bar{x}^B)^2 M_0^C M_0^B}{M_0^C + M_0^B}. \quad (5)$$

Update formulas for moments in directions y and z are analogous and separately computable.

3. OCCUPANCY MAPS

Spatial distribution within a point cloud is applied as a guide to ME, restricting more costly voxel matching procedures and reducing execution time. For such, the local shape descriptors of the point cloud are summarized within an occupancy map. The map consists of a voxel space of dimensions N^3. Each point within the map $O(x, y, z)$ contains a shape descriptor for a local cube C of dimension L and origin (x, y, z).

The occupancy map may be efficiently computed and stored prior to ME. The voxel space containing the point cloud is scanned incrementally. For each new point, 3D moments of the local cube are updated as described in Sec. 2. For efficient map storage we consider only scalar-valued descriptors and propose the usage of (i) size or (ii) the second

central moment along the axis of minimum variance. Note that point clouds are formed by points on the surface of the captured objects which, at a local level, may be assumed as approximately planar patches. As such they present an axis of minor variance, although not necessarily fully aligned with x, y or z. The second descriptor selects among these latter axes the one displaying minimum overall voxel dispersion with respect to local cube centroid, adopting this scalar value for local shape description. The considered descriptors are respectively registered in the following maps:

$$O_0(x,y,z) = M_0^C \tag{6}$$

$$O_2(x,y,z) = \min(\mu_{200}^C, \mu_{020}^C, \mu_{002}^C). \tag{7}$$

The motion estimator compares the local shape statistic of the current source cube $O_i^{src}(x,y,z)$ to those contained in the map and restricts the search space to target cubes presenting similar statistics, i.e., within a tolerance range defined by a threshold T: $(1-T)O_i^{src}(x,y,z) \leq O_i^{tgt}(x',y',z') \leq (1+T)O_i^{src}(x,y,z), i \in \{0,2\}$.

4. EXPERIMENTAL RESULTS

Speedup for ME was tested on the publicly available point cloud data sets *Andrew*, *David*, *Phil*, *Ricardo*, *Sara*, *Man* [13] and *Loot* [8]. The first five sequences are upper body and the others, full body human subjects. All have spatial resolution of $512 \times 512 \times 512$ voxels and are furnished with RGB color attributes.

A cube-matching criterion similar to that of [7] is adopted wherein nearest neighbor correspondences are determined between cube voxels. The average Euclidean distance δ_g and average color distance δ_c, in Y-channel, between correspondences are combined in $\delta = \delta_g + 0.35\delta_c$. The final matching distance is symmetric and considers the maximum δ among source-to-target and target-to-source cubes. For each sequence we use frame #10 as the source and #9 as the target, cube dimension $L = 8$ and search space dimension $S = 15$.

Distortion between predicted and source point cloud is measured with two peak signal-to-noise ratio metrics [7, 14], both of which contemplate geometric and color differences. The first metric (PSNR-P) orthographically projects the point cloud voxels onto the planes defined by the faces of a surrounding cube of dimension 512, forming 2D images. From each of the 6 projected image pairs, originating from the predicted and the source voxel sets, an overall mean square error (MSE) in the Y-channel color component is determined and PSNR-P $= 10\log(255^2/\text{MSE})$. The second distortion metric (PSNR-Y-G) determines, for each voxel of the source cloud, a nearest neighbor within the predicted cloud. Once all correspondences are established, a matching distance δ is calculated and PSNR-Y-G $= 10\log(255^2/\delta)$.

Fig. 3: Data points with sizes of the source and corresponding target cube (determined through full search ME) for each of the 3410 source cubes in *Man*. The linear regression model is in red.

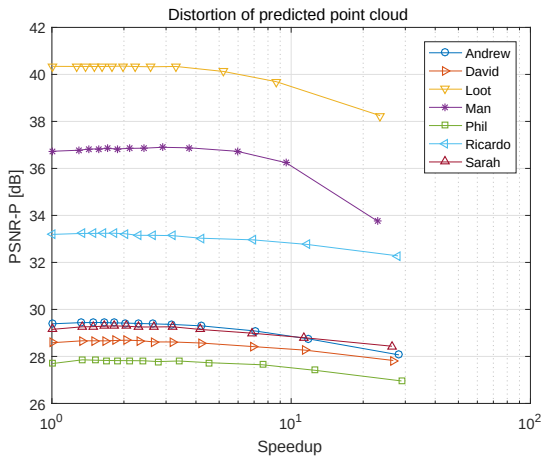
Speedups are measured as ratios between the full search space and the reduced search spaces, the latter using only target cubes within tolerance ranges defined in Sec. 3. We exclude from the full search space count the trivial case where the target cube is empty. Tolerance thresholds T, in percentages, are selected from $\{\infty, 100, 90, ..., 10, 5, 1\}$.

In Fig. 3 we illustrate the relation between the size of each source cube and that of its target cube, determined through full search ME. The approximately one-to-one relationship indicates the potential of this descriptor as a means for distinguishing incompatible candidates. This relationship holds true for other sequences and for the second moment descriptor of (7).

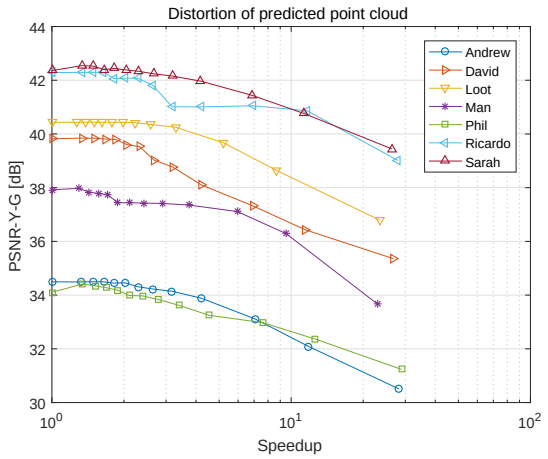
Predicted frame distortions in terms of PSNR-P and PSNR-Y-G are presented in Fig. 4 for various sequences and speedups achieved with the size-based occupancy map of (6). Distortion of full search ME prediction is indicated at speedup 1. A speedup of 2 guarantees a maximum distortion dropoff, relative to full search ME, among all sequences and metrics of less than 2%. A speedup of 5 introduces average PSNR-P and PSNR-Y-G dropoffs of 0.2% and 2.5%, respectively. At speedups of 10, average dropoffs are 1.3% and 4.8% are attained. In terms of PSNR-P, the worst performance among sequences, at speedups 2, 5 and 10, is for *Loot* with dropoffs of 0.0%, 0.5% and 1.9%, respectively. In terms of PSNR-Y-G, the worst performance at speedups 2, 5 and 10 is for *Man* with 1.3% and *David* with more significant dropoffs of 4.8% and 7.8%, respectively.

The motion vector fields, akin to optical flow, resulting from the ME speedup framework are compared to those resulting from full search through the average absolute error in flow endpoints [15]. Results are shown in Fig. 5. Small differences occur on the onset of speedup although these are not necessarily correlated to prediction quality.

The potential of the second order central moments along



(a)



(b)

Fig. 4: (a) PSNR-P and (b) PSNR-Y-G distortions as a function of speedup using size-based occupancy maps for various sequences.

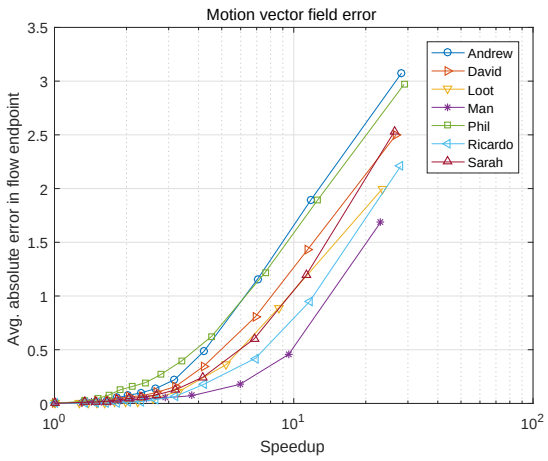


Fig. 5: Motion vector field errors [15], with respect to full search ME, as a function of speedup using size-based occupancy maps for various sequences.

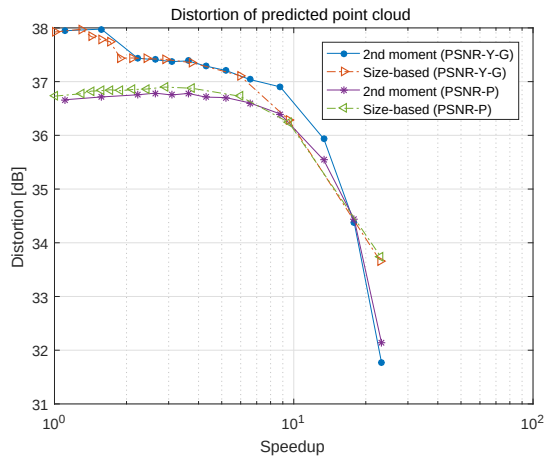


Fig. 6: PSNR-P and PSNR-Y-G distortions as a function of speedup using size and second order central moments as occupancy descriptors for the *Man* sequence.

the axes of minimum variance as occupancy descriptors is illustrated in Fig. 6. Performance in terms of PSNR-P and PSNR-Y-G for *Man* is similar to those of the size-based descriptor. Comparable results are achieved for the other sequences as well.

5. CONCLUSIONS

We proposed the use of 3D moment-based shape descriptors as a means of efficiently characterizing local spatial distribution within point clouds in order to speedup block-based ME. By consulting an occupancy map containing information regarding size or other higher-order statistics of a target frame, candidate cubes may be eliminated from the search space prior to more costly cube-matching evaluations. Descriptors forming the occupancy map are efficiently computed with one-pass update formulas and stored as scalar-values for multiple, subsequent references. Results show that a speedup of 2 implies a maximum distortion dropoff of less than 2% for the adopted PSNR-based metrics, relative to distortion of the prediction attained from full search ME. Speedups of 5 and 10 are achievable with small average distortion dropoffs, less than 3% and 5%, respectively. Future work includes further improvement of computational efficiency, the usage of local covariances for planar modeling of greater precision and the improvement of storage and compression of occupancy maps containing vectorial descriptors of greater complexity.

6. REFERENCES

[1] J. Kammerl, N. Blodow, R. B. Rusu, S. Gedikli, M. Beetz, and E. Steinbach, "Real-time compression of point cloud streams," in *IEEE Int. Conf. on Robotics and Automation*, May 2012.

[2] C. Loop, C. Zhang, and Z. Zhang, "Real-time high-resolution sparse voxelization with application to image-based modeling," in *Proc. High-Performance Graphics Conf.*, Jul. 2013.

[3] C. Zhang, D. Florencio, and C. Loop, "Point cloud attribute compression with graph transform," in *Proc. IEEE Int. Conf. on Image Process.*, Oct. 2014.

[4] R. L. de Queiroz and P. A. Chou, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Trans. Image Process.*, vol. 25, no. 8, Aug. 2016.

[5] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3D point cloud sequences," *IEEE Trans. Image Process.*, vol. 25, no. 4, Apr. 2016.

[6] R. Mekuria, K. Blom, and P. Cesar, "Design, implementation and evaluation of a point cloud codec for tle-immersive video," *Trans. Circuits Syst. Video Technol.*, vol. 27, no. 4, Apr. 2017.

[7] R. L. de Queiroz and P. A. Chou, "Motion-compensated compression of dynamic voxelized point clouds," *IEEE Trans. Image Process.*, vol. 26, no. 8, Aug. 2017.

[8] E. d'Eon, B. Harrison, T. Myers, and P. A. Chou, "8i voxelized full bodies - a voxelized point cloud dataset," in *ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document WG11M40059/WG1M74006*, Geneva, Switzerland, Jan. 2017.

[9] A. M. Tourapis, "Enhanced predictive zonal search for single and multiple frame motion estimation," *Proc. SPIE Visual Comm. and Image Process.*, vol. 4671, 2002.

[10] F. A. Sadjadi and E. L. Hall, "Three-dimensional moment invariants," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 2, no. 2, Mar. 1980.

[11] P. Pébay, "Formulas for robust, one-pass parallel computation of covariances and arbitrary-order statistical moments," Tech. Rep. SAND2008-6212, Sandia National Laboratories, 2008.

[12] T. F. Chan, G. H. Golub, and R. J. LeVeque, "Updating formulae and a pairwise algorithm for computing sample variances," Tech. Rep. STAN-CS-79-773, Department of Computer Science, Stanford University, 1979.

[13] C. Loop, Q. Cai, S. O. Escolano, and P. A. Chou, "Microsoft voxelized upper bodies - a voxelized point cloud dataset," in *ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document m38673/M72012*, Geneva, Switzerland, May 2016.

[14] R. L. de Queiroz, E. Torlig, and T. A. Fonseca, "Objective metrics and subjective tests for quality evaluation of point clouds," in *ISO/IEC JTC1/SC29/WG1 input document M78030*, Rio de Janeiro, Brazil, Jan. 2018.

[15] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *Int. Journal of Computer Vision.*, vol. 92, no. 1, Mar. 2011.