

AUTOMATED ALIGNMENT OF LOCAL POINT CLOUDS IN DIGITAL BUILDING MODELS

T. Kaiser¹, C. Clemen¹, H.-G. Maas²

¹ Faculty of Spatial Information, Dresden University of Applied Sciences, Friedrich-List-Platz 1, D-01069 Dresden, Germany – (tim.kaiser, christian.clemen)@htw-dresden.de

²Institute of Photogrammetry and Remote Sensing, TU Dresden, Germany - hans-gerd.maas@tu-dresden.de

Commission II

KEY WORDS: Structure from Motion, Point Cloud, Registration, Building Information Modeling

ABSTRACT:

For the correct usage and analysis within a BIM environment, image-based point clouds that were created with Structure from Motion (SfM) tools have to be transformed into the building coordinate system via a seven parameter Helmert Transformation. Usually control points are used for the estimation of the transformation parameters. In this paper we present a novel, highly automated approach to calculate these transformation parameters without the use of control points. The process relies on the relationship between wall respectively plane information of the BIM and three-dimensional line data that is extracted from the image data. In a first step, 3D lines are extracted from the oriented input images using the tool Line3D++. These lines are defined by the 3D coordinates of the start and end points. Afterwards the lines are matched to the planes originating from the BIM model representing the walls, floors and ceilings. Besides finding a suitable functional and stochastic model for the observation equations and the adjustment calculation, the most critical aspect is finding a correct match for the lines and the planes. We therefore developed a RANSAC-inspired matching algorithm to get a correct assignment between elements of the two data sources. Synthetic test data sets have been created for evaluating the methodology.

1. INTRODUCTION

A BIM-Model depicts a complete digital representation of a building. The model is used as the extensive information source during the complete life cycle of the real world building beginning with the planning and construction, continuing with the operation and eventually stops with the demolition. During these different phases, it will be possible and almost inevitable that deviations between the as-built and the originally as-planned state will occur. Deviations can be as early as in the building phase. This means that the original created BIM-Modell needs to be updated frequently to represent the actual state of the building.

Structure-from-Motion (SfM) techniques allow reconstructing an object from a collection of images (e.g. Westoby et al. 2012). In conjunction with multi-view stereo algorithms dense point clouds can be further processed to meshes representing the geometry of the object. Due to its relatively low capturing and processing costs SfM became very popular in the geospatial domain in recent years.

With the construction sector in mind, such point clouds can contain the required geometric information for updating the building model for getting the actual as-built state. Especially using local points of closed rooms inside a building could provide a range of possible applications. They could be used for the measurement, modeling and documentation of subsequently build in structures in the room (e.g. added radiators or cable channels). Besides of that little damages like cracks could be registered and located accurately. So overall using the extracted building information from the BIM-Model in conjunction with the local point cloud enables a comparison of the as-planned and the as-built state as well as updating the as-built model.

However, there is the challenge of having two different coordinate systems when combining the two data sources. During the creation of a SfM point cloud an arbitrary coordinate system

is used. For a usage within the presented use case, the point cloud must be transformed into the building coordinate system based on a seven parameter Helmert-Transformation. This usually requires the manual measurement of ground control points with known coordinates in both systems. However, this task is time-consuming, error-prone and repetitive for big data sets.

Our presented approach does not depend on a manual measurement of points but rather uses geometric relationships between specific BIM-objects and extracted line structures from the captured images to compute the transformation parameters. This approach enables an automatic room-based transformation of the local point cloud into the building coordinate system for further analysis.

2. RELATED WORK

In recent years multiple case studies (e.g. Tuttas et al. 2017; Golparvar-Fard et al. 2015) were performed using BIM-models and photogrammetric acquired point clouds for the progress monitoring of a construction site. By comparing the as-built with the as-planned state deviations can be noted early in construction stages so that it is possible to avoid further mistakes which could lead to delays or increased costs. However, the mentioned papers cover a complete construction site and not only a single room like our system intends to do. Besides of that it is also required to transform the acquired point clouds into the building coordinate system for the comparison. This is achieved by measuring Ground Control Points (GCP) for calculating the transformation parameters.

For establishing a more automatic registration process of a complete point cloud to a given BIM-model Kim et al. (2013) waived to use GCPs. They used a combination of a Principal Component Analysis and an Iterative Closest Point Algorithm to perform a successful registration between the point cloud and the building model. The achieved results are depending on certain

circumstances though, so that further research has to be done for a universal application of this method.

In addition to an image based reconstruction terrestrial laser scanners are often used for the generation of building point clouds. However, it is still necessary to perform a transformation of the point cloud into the building coordinate system. The only difference is that point clouds from laser scanners are up to scale so that it is not required to estimate the scale factor. Due to the high point density in such point clouds plane-based transformation methods are often used. Planes are searched in the scanned point cloud and are related to extracted planes from the building model. The challenging step is the correct matching of corresponding planes in the point cloud and in the model. This often requires manual involvement (Bosché 2012).

3. PROPOSED CONCEPT AND IMPLEMENTATION

The standard approach for transforming local point clouds into a global coordinate system is to calculate the transformation parameters with an adjustment calculation using control points (CPs). Finding and setting up such CPs can become a challenging task. Due to the rapidly changing nature of a construction site or a building in general, CP locations must be set up at a location where they are durable and are visible from a wide point of view in every construction and lifecycle phase. It has to be taken care of this when establishing the CPs on the construction site. That means that expert knowledge is required for the installment of the CPs. Since a BIM-model of the building is already given in our specific use-case we want to renounce from using CPs in favor of a more automated workflow. The idea is to use the relationship of extracted three dimensional line structures and planes resulting from the BIM-model for estimating the transformation parameters.

3.1 General Workflow of Concept

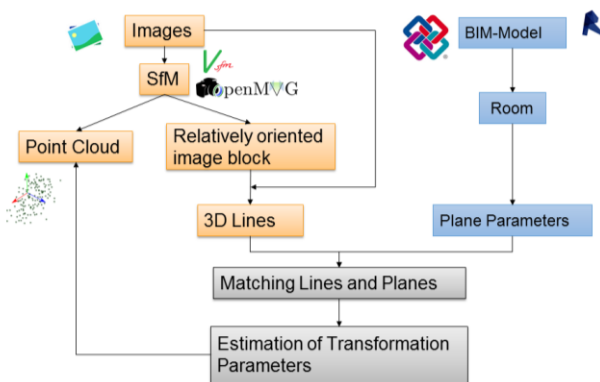


Figure 1: Overview on the alignment process

The workflow (Figure 1) starts with the capturing of the images of the corresponding room. An arbitrary SfM pipeline is run afterwards for generating the sparse point cloud and getting the interior and exterior orientation of the images. This information is then used for extracting the 3D line structures from the images using the tool Line3D++ (Hofer et al. 2017). For the later analysis, the dense point cloud is required. However, the dense point cloud is not directly needed for the transformation parameter estimation process but is then transformed in the building coordinate system using the results of the estimation.

The building coordinate system is the target system in the alignment process. This requires retrieving the corresponding room's plane parameters from the available BIM-Model. Since

the presented concept is intended to work within a special separated area of a building, the plane parameters of the affected boundaries must be extracted from the model. Having gathered both the line and plane data the matching of these two data sets can be done. Finally, this information is used for the calculation of the transformation parameters.

3.2 Line Capturing using Line3D++

For capturing three dimensional line structures the software tool Line3D++, is used. Line3D++ provides the coordinates of the start and end points of the line structures. The coordinates are defined in the same coordinate system as the point cloud (Figure 2).

In a first step, Line3D++ is extracting two dimensional line segments in each image. It then matches the possible candidates across the other visual neighbors and uses epipolar geometry constraints to get the three dimensional start and end point coordinates of the straight line segments. For a detailed in deep explanation of the algorithm we refer to the corresponding paper.

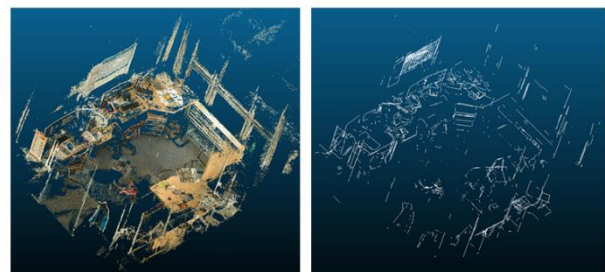


Figure 2. Generated point cloud of a room (left) and extracted 3D line structures (right)

Besides of the (unordered) image collection, Line3D++ requires several other input information to work. This mainly concerns the interior and exterior orientation of the images. The camera parameters are used for undistorting the images whereas the exterior orientation and the sparse point cloud are used to determine which images are visual neighbors. In order to get this information a SfM process must be run beforehand. Line3D++ therefore supports various input data formats from different open source SfM pipelines such as openMVG, VisualSfM or Colmap.

3.3 Plane Capturing from the BIM-Model

The major advantage of a BIM-Model compared to a building model created with CAD technology is that it not only contains the geometric representation but rather enriches the geometry with a multitude of semantic information. Such a model consists of several objects like walls, doors or windows that are all linked to each other and the remaining building parts. In the CAD world a wall just consists of its limiting set of lines whereas in BIM a wall is an object with specified attributes and relations to other objects (Clemen and Gründig, 2006). To sum it up, BIM provides a much deeper semantic integration as CAD and especially a topologically correct representation of the building. Due to these characteristics of such a model, it becomes possible to select all affected walls of the room for example by specifying the according room number.

The following procedure could be applied for extracting the relevant information from the main components (i.e. the walls, floor and ceiling) of the building model. The proposed concept assumes that building floors and ceilings are horizontal and walls are vertical with respect to the building coordinate system. Due

to the highly dynamic changes in BIM authoring tools and data formats the algorithm is explained in a general form and has to be adapted to the specific tools and data. The basic output of the algorithm is a list of tuples that consists of the normal vector of the plane and a point located on the plane for every plane forming the room. This plane parametrization is equal to the Hesse normal form.

The first step is to select the room from the building model as a separate object. This room belongs to a story that is used for deriving the floor and ceiling that in turn enable getting the upper and lower plane of the room (Figure 3).

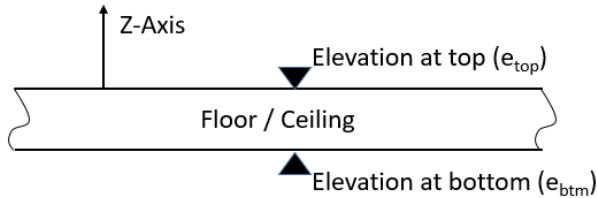


Figure 3: Side view of a floor or ceiling element

The normal of the room's floor plane is the Z-Axis of the building coordinate system. For a point located on the corresponding plane the elevation at top is multiplied with the Z-Axis. Similar to the lower plane the ceiling's normal vector is the negative Z-Axis and for getting a plane point the elevation at bottom is multiplied with the Z-Axis.

$$\begin{aligned} P_{top} &= \{ \vec{z}_0, e_{btm} \text{ Ceiling } \hat{z} \} \\ P_{btm} &= \{ -\vec{z}_0, e_{top} \text{ Floor } \hat{z} \} \end{aligned} \quad (1)$$

where P_{top}, P_{btm} = Point on Floor / Ceiling
 \vec{z}_0 = Normalized Vector of Z-Axis
 e_{btm}, e_{top} = Elevation at Top / Bottom of Plane

For retrieving the remaining planes, a consecutive list of the enclosing wall segments is needed. Every wall segment is defined by an axis (through the middle of the wall) between the starting point A and the end point B and the overall thickness t of the wall segment (Figure 4). It must be ensured that the directions of the axes point counter clockwise around the room.

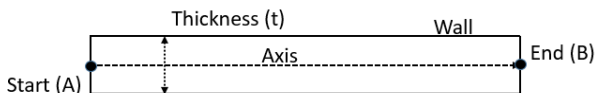


Figure 4: Topview of a wall element.

The normal vector of a wall plane is calculated as follows:

$$\begin{aligned} \vec{n} &= \vec{z} \times (B - A) \\ \vec{n}_0 &= \frac{\vec{n}}{|\vec{n}|} \end{aligned} \quad (2)$$

where \vec{n}_0 = Normalized Normal Vector
 \vec{z} = Z-Axis
 A, B = Starting and End Point of Wall Axis

3.4 Estimation Algorithm and Line Plane Assignment

This section describes the process for aligning the generated point cloud and the building model. A classical least squares

method is applied for estimating the unknown transformation parameters of the Helmert Transformation. These are:

- Three rotations around the coordinate axes (equal to the nine elements of the corresponding rotation Matrix R)
- Three components of the translation vector \vec{t}
- A scale parameter m

3.4.1 Functional Adjustment Model

For solving the adjustment problem, geometric relationships between the extracted lines and the room planes are used in the observation equations. A BIM-model allows to extract plane parameters that can be converted in the coordinate form describing the plane in the form:

$$ax + by + cz = d \quad (3)$$

For lines that are directly located on a wall of the room the rotated direction vector \vec{u} must be perpendicular to the normal vector $\vec{n} = [a \ b \ c]^T$ of the wall plane in the common coordinate system. This means that the scalar product between these two vectors must be zero. Furthermore must the start and end points of the rotated, translated and scaled line be lying in the corresponding plane. Similar to Clemen and Gielsdorf (2008) the two constraints are defined by the following two (pseudo-) observation equations:

$$l + v = \langle R * \vec{u}, \vec{n} \rangle = 0 + v \quad (4)$$

$$l + v = m * \langle (R * \vec{s} + \vec{t}), \vec{n} \rangle - d = 0 + v \quad (5)$$

where R = Rotation Matrix
 \vec{t} = Translation Vector
 m = Scale Parameter
 l = Observations
 v = Residuals
 \vec{u} = Direction Vector of Line
 \vec{n} = Normal Vector of Plane
 \vec{s} = Position Vector

In addition, it is necessary to introduce restrictions for getting a valid rotation matrix. One of the characteristics of a rotation matrix is that it is an orthonormal basis. Therefore, the three column vectors of the matrix must be of length 1 and the column vectors must also be orthogonal to each other. These demands can be achieved using the following restrictions:

$$\begin{aligned} w_1 &= r_{11}^2 + r_{12}^2 + r_{13}^2 - 1 = 0 \\ w_2 &= r_{21}^2 + r_{22}^2 + r_{23}^2 - 1 = 0 \\ w_3 &= r_{31}^2 + r_{32}^2 + r_{33}^2 - 1 = 0 \\ w_4 &= r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} = 0 \\ w_5 &= r_{12}r_{13} + r_{22}r_{23} + r_{32}r_{33} = 0 \\ w_6 &= r_{11}r_{13} + r_{21}r_{23} + r_{31}r_{33} = 0 \end{aligned} \quad (6)$$

where $r_{11} - r_{33}$ = Elements of the Rotation Matrix
 $w_1 - w_6$ = Restriction Equations

The transformation parameters are estimated using the following Gauss-Markov model and a least squares approach.

$$\begin{bmatrix} A^T P A & B \\ B^T & 0 \end{bmatrix} * \begin{bmatrix} \hat{x} \\ k \end{bmatrix} = \begin{bmatrix} A^T P l \\ w \end{bmatrix} \quad (7)$$

where \hat{x} = Transformation Parameters
 B = Linearized Restriction Equations
 P = Weight Matrix
 k = Lagrangian Multipliers

w = Vector of Inconsistencies
 A = Design Matrix
 l = Observations

3.4.2 Stochastic Adjustment Model

Since multiple input measurements are involved in the estimation process it is important to formulate a proper stochastic model for assessing the results. We assume that the measured pixel coordinates are of equal accuracy and are not correlated. However, it is important to know the standard deviation of each line contributing to the adjustment process. That is why the standard deviation and X,Y,Z-correlation for every three dimensional start and end point is calculated the following way.

Since a SfM process has been run, the interior and exterior orientation of the images is known. This enables to retrieve the pixel coordinates of the line defining end points using the projection matrix Prj (since Line3D++ is using the distortion free images no additional distortion parameters are considered). This makes it possible to set up two observation vectors for each 3D line. The vector L_{mes} comprises the pixel coordinates outputted by Line3D++ whereas the vector L_{prj} contains the pixel coordinates calculated by projecting the 3D locations of the start and end points on the image plane using the corresponding projection matrix.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Prj \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (8)$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

where: x, y = Pixel Coordinates of Line Endpoint
 f_x, f_y = Focal Length
 c_x, c_y = Coordinates of Projection Center
 $r_{11} - r_{33}$ = Elements of Rotation Matrix
 $t_1 - t_3$ = Translation Vector

By subtracting both vectors the residuals v can be calculated. The residuals enable to determine the standard deviation of the unit weight $\hat{\sigma}_0$.

$$v = L_{mes} - L_{prj} \quad (9)$$

$$\hat{\sigma}_0 = \sqrt{\frac{v^T * P * v}{n - u}} \quad (10)$$

where: v = Residuals
 L_1, L_2 = Observation Vectors
 n = Number of Measurements
 u = Number of Unknowns
 P = Weight Matrix

Next, the covariance matrix of the homogenous object coordinates is calculated. For computation of the normal equation matrix the projection matrix is used as design matrix.

$$C_{xx} = \sqrt{\frac{v^T * P * v}{n - u}} * (Prj^T * Prj)^{-1} \quad (11)$$

where: C_{xx} = Covariance Matrix of Lines
 $\hat{\sigma}_0$ = Unit Weight

Prj = Projection Matrix for both Points

By following the law of propagation of errors, it is possible to get the covariance matrix of the line coordinates. With the aid of this covariance matrix the stochastic information of the extracted 3D lines can be considered in the functional model.

3.4.3 Line Plane Assignment

The functional model requires knowledge about the line to plane assignment. However, there is no such linking information between the lines and planes available. Establishing a correct assignment of the used lines to the corresponding planes is the crucial point for the complete transformation process.

Without further data all theoretical combinations of lines and planes could be possible. For solving this assignment problem within a reasonable amount of time it is necessary to reduce the big number of possibilities since the usage of a brute force approach is too costly.

By applying a filtering step, invalid combinations are sorted out before getting into the adjustment calculation. The minimal configuration (that is the minimum required number of observations for the solution of the adjustment problem) consists of four non-coplanar line plane assignments. We therefore calculate the triple product for each possible combination of three lines (respectively their direction vectors). In Euclidean space the triple product represents the volume of the parallelepiped defined by the three vectors. Only the combinations with a big volume are kept as the assumption is that the bigger the spanned volume the more orthogonal the vectors are located to each other. This in turn would mean that they are on different planes supporting the demand for four non-coplanar lines. This filtering step is also performed for the planes using the respective normal vectors.

After the filtering step the actual assignment procedure continues following a RANSAC-based (Fischler, Bolles, 1981) algorithm. The implemented approach randomly selects one of the filtered line triples and assigns the triple to one of the plane triples. For completing the minimal configuration a fourth non coplanar line is added. Since no information is available which line belongs to which plane all planes have to be considered that are not yet included in the first randomly picked plane triple. With the complete minimal configuration the adjustment calculation is run. The resulting parameters are then used for transforming the lines of the minimal configuration into the building coordinate system. Afterwards the summed distance between the midpoints of the lines and the assigned plane is calculated and saved. This procedure is repeated for a defined number of iterations depending on the number of affected planes and lines.

After completing all iterations, the transformation parameters of the combination with the smallest distance sum is chosen. The parameter set is used to transform the complete line data set into the building coordinate system. For every line, the plane with the smallest distance is selected afterwards. If the distance is below a defined threshold (that is set for eliminating wrong line plane assignments) the line is assigned to the corresponding plane. All lines that were assigned to a plane are consequently used for the final estimation of the transformation parameters. The following flow chart (Figure 5) summarizes the estimation process:

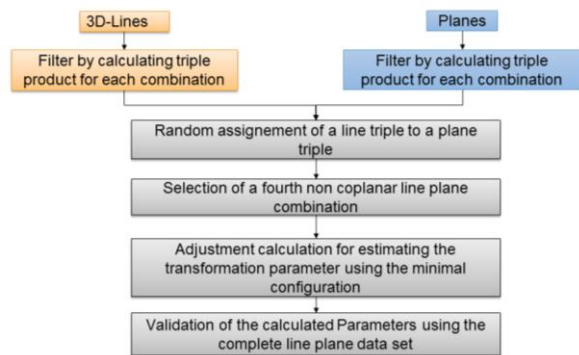


Figure 5: Workflow for the transformation parameter estimation

4. VALIDATION OF THE PRESENTED APPROACH

For a quick validation of the transformation estimation process we generated a set of synthetic test datasets (Figure 6). Multiple rooms with different numbers of planes were created using a CAD Software. Lines were placed on all planes respectively walls, the floor and the ceiling. We could successfully transform the lines into the building coordinate system.

In order to have a more realistic data source we randomly added noise to the lines that originally are placed perfectly into the planes. For the correct application of the transformation process it is also very important that it is robust to outliers. In this context outliers are such lines that are detected by Line3D++ but originally are not placed on a wall but for example in the middle of the room or on a tabletop where no plane parameters are known. For simulating this case we intentionally added fake lines to the data set. Experiments with this data have shown that for a successful parameter estimation up to 13% outliers can be contained in the input data.

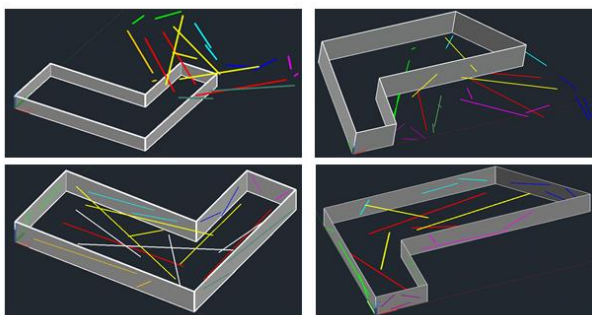


Figure 6: Synthetic datasets used for validating the concept

5. CONCLUSION

In this paper we presented a novel concept for the automatic transformation of locally created point clouds into a coordinate system of a building model. By using additional information from the given building model in conjunction with the extracted 3D line data it becomes possible to correctly register the point cloud without having to measure additional GCPs. The proposed system does not require special hard- or software and therefore could be used by a wide target audience.

The next step will be to evaluate the implemented workflow with real data. It will be interesting to see if the implemented algorithm is robust enough in the current state to deal with a potentially high number of outliers.

ACKNOWLEDGEMENTS

The European Social Fund (ESF) and the Free State of Saxony (funding no. 100316427) financially support this research project.



REFERENCES

- Bosché, F., 2012: Plane-based registration of construction laser scans with 3D/4D building models. In: *Advanced Engineering Informatics* 26 (1), 90–102.
- Clemen, C.; Gielsdorf, F., 2008: *Architectural Indoor Surveying. An Information Model for 3D Data Capture and Adjustment*. In: *American Congress on Surveying and Mapping (ACSM) 2008*. Spokane, WA, USA.
- Clemen, C.; Gründig, L., 2006: *The Industry Foundation Classes - Ready for Indoor Cadastre?* In: *Proceedings of the XXIII FIG Congress*. München. International Federation of Surveyors.
- Fischler, M. A.; Bolles, R. C., 1981: *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography*. In: *Commun. ACM* 24 (6), 381–395.
- Golparvar-Fard, M.; Peña-Mora, F.; Savarese, S., 2015: *Automated Progress Monitoring Using Unordered Daily Construction Photographs and IFC-Based Building Information Models*. In: *J. Comput. Civ. Eng.* 29 (1), S. 4014025.
- Hofer, M.; Maurer, M.; Bischof, H., 2017: *Efficient 3D scene abstraction using line segments*. In: *Computer Vision and Image Understanding* 157, 167–178.
- Kim, C.; Son, H.; Kim, C., 2013: *Fully automated registration of 3D data to a 3D CAD model for project progress monitoring*. In: *Automation in Construction* 35, 587–594.
- Tuttas, S.; Braun, A.; Borrmann, A.; Stilla, U., 2017: *Acquisition and Consecutive Registration of Photogrammetric Point Clouds for Construction Progress Monitoring Using a 4D BIM*. In: *PFG* 85 (1), 3–15.
- Westoby, M. J.; Brasington, J.; Glasser, N. F.; Hambrey, M. J.; Reynolds, J. M. (2012): 'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications. In: *Geomorphology* 179,